Un cadre logique pour la génération d'arguments

Geoffroy Aubry

Université de la Méditerranée
Aix-Marseille II
Faculté des sciences de Luminy

École doctorale de mathématiques
et d'informatique de Marseille

THÈSE

en informatique

présentée par

Geoffroy Aubry

pour obtenir le grade de docteur de l'université Aix-Marseille II

Un cadre logique pour la génération d'arguments

Soutenue publiquement le 13 décembre 2005 devant le jury composé de :

Philippe Besnard
Didier Dubois
Robert E. Mercer
Vincent Risch
Pierre Siegel
Francesca Toni

Année ␣␣␣␣ N° ␣␣␣␣␣␣␣␣␣␣

préparée au sein du

Laboratoire des Sciences de l'Information et des Systèmes
UMR CNRS 6168

Remerciements

Je tiens à exprimer ma profonde gratitude à Vincent Risch qui a dirigé mes recherches tout au long de cette thèse. Pour toutes ces discussions passionnées et amicales, ces nombreux conseils et pour toute l'attention et la gentillesse qu'il a su me prodiguer, jamais je ne le remercierai assez.

Je remercie particulièrement Francesca Toni et Didier Dubois qui m'ont fait l'honneur d'être les rapporteurs de ce mémoire, dans des conditions difficiles.

Merci à Pierre Siegel qui a accepté d'être mon directeur de thèse temporaire !

Je remercie également Philippe Besnard et Robert E. Mercer qui ont bien voulu se déplacer pour ma soutenance.

Je tiens à remercier profondément Pierre Basso, Philippe Jégou, Nicolas Prcovic et Cyril Terrioux pour leur accueil, leurs précieux conseils et leur disponibilité.

Un grand merci à Norbert Giambiasi à qui je dois une amélioration de mes conditions de travail, à plusieurs reprises.

Je remercie Sylvie Risch pour ses innombrables encouragements si chaleureux.

Merci à tous mes amis pour leur soutien.

Merci à ma famille qui m'a aidé plus qu'elle ne le croit.

Table des matières

Liste des tableaux

Table des figures

Introduction

La capacité à mener un raisonnement est l'une des multiples expressions de ce que l'on nomme intelligence. Tenter de l'appréhender par le biais du calcul nous fait entrer de plein pied dans l'Intelligence artificielle.

L'Intelligence artificielle — ou l'IA — est une discipline de l'informatique qui s'intéresse à l'automatisation de tâches demandant des connaissances et des processus de raisonnement complexes. Elle est depuis fort longtemps engagée dans le défi de la modélisation du raisonnement de sens commun ([MH87]), faisant intervenir des informations incomplètes, incertaines ou éventuellement inconsistantes.

Rapidement, la logique classique a été considérée comme un outil privilégié des chercheurs pour la formalisation et l'exploitation des connaissances. Toutefois, elle reste inadéquate pour la représentation et le raisonnement avec des connaissances de sens commun. En particulier l'inférence classique ne peut prendre en compte des informations partiellement incohérentes, ce que l'intelligence humaine paraît maîtriser. Les limites de la logique classique ont donné lieu à de nombreuses extensions appelées logiques non classiques. Parmi ces extensions figurent des logiques dites *non-monotones* ([Bob80]) qui ne satisfont donc pas la propriété de monotonie de la logique classique. Cette propriété exprime le fait qu'un résultat déduit n'est jamais remis en cause, ce qui est incompatible avec l'incertitude liée aux connaissances mises en jeu : dans le raisonnement de sens commun, les conclusions obtenues ne sont que plausibles et donc éventuellement *révisables* à la lumière de nouvelles informations. Ce « raisonnement révisable » est très fréquent puisqu'il permet de tirer des conclusions même en l'absence d'informations complètes.

L'*argumentation* est une toute autre approche du raisonnement révisable, que l'on pourrait qualifier de locale. Intuitivement, cette approche est basée sur la justification d'une conclusion plausible par des arguments en faveur de celle-ci. L'argumentation est un modèle prometteur pour raisonner avec des connaissances incertaines ou incohérentes. Ce modèle est essentiellement fondé sur la construction d'*arguments* et de *contre-arguments*, la comparaison de ces différents arguments puis finalement la sélection de ceux jugés acceptables. Il est à noter que la construction d'arguments est monotone, un argument reste toujours un argument même si de nouvelles connaissances sont ajoutées. La non-monotonie apparaît en termes d'interactions entre arguments conflictuels. C'est l'ajout de nouvelles connaissances qui peut donner lieu à de nouveaux contre-arguments. Les arguments et les *conflits* entre arguments mis en jeu dans le processus d'argumentation peuvent être représentés dans ce que l'on appelle un *système argumentatif*. Il est important de remarquer que les systèmes argumentatifs ne s'intéressent pas à la vérité des propositions, mais seulement à leur justification.

L'argumentation est un domaine relativement jeune, mais suffisamment mature pour fournir des solutions à d'autres secteurs. Elle est notamment devenue très populaire dans le domaine juridique ([LNOM93, Pra93, Sar94, Gor95, PS96, Pra97]). Récemment, des systèmes argumentatifs ont aussi été appliqués au domaine médical ([DFK96]), à la négociation entre agents ([Syc89, Syc90, PSJ98, CML00]) et à la prise de décision ([FAM96, GGTS00]). L'argumentation est également étudiée dans d'autres disciplines comme la langue et la cognition ([AD85a, AD85b, KP89, KP93]) : psychologie, linguistique ou philosophie sont concernées par ce sujet. L'argumentation possède donc un large éventail d'applications. Elle peut être appliquée à toute forme de raisonnement impliquant des informations conflictuelles.

Cette thèse a pour objectif la génération automatique de nouveaux arguments au sein d'une discussion entre deux agents. C'est un aspect rarement abordé directement. Notons que l'on ne se restreint pas uniquement à la construction de contre-arguments mais souhaitons produire également des arguments de soutien ou d'accord, c'est-à-dire des arguments du type « Je suis d'accord avec vous, parce que ... ». Donnons une idée du processus de génération. Nous partageons la conception d'un argument vu comme un couple support-conclusion, où le support est la raison de croire en la conclusion. Lors d'une discussion, deux *agents* vont s'échanger des arguments. Le premier agent en avance un, c'est ensuite au tour du second de générer un argument en retour. La conclusion de celui-ci est à choisir en fonction des *attitudes* du deuxième agent par rapport au premier argument : s'il n'est pas d'accord sur une partie de l'argument, il peut orienter sa conclusion en ce sens, ou au contraire souligner grâce à celle-ci un point d'entente. Des *opérateurs de confrontation* doivent permettre de déterminer les attitudes de l'agent vis-à-vis des différentes parties de l'argument extérieur. Une fois la conclusion du futur argument arrêtée, il faudra générer son support. À cet effet nous avons élaboré le concept de *réponse* : une réponse est l'ensemble d'informations — de formules — en possession de l'agent telles que sans elles, l'attitude de cet agent sur le point ciblé n'aurait pas été la même. La réponse est le support de l'argument et la motivation de la conclusion. Pour produire ces réponses, tout est fait comme si l'agent cherchait à se convaincre : sans entrer dans le détail, on inverse les rôles, et les opérateurs de confrontation vont exhiber les parties des connaissances de l'agent suffisantes à la justification de l'attitude de ce dernier.

La constitution d'une théorie logique permettant la représentation d'un système argumentatif est un problème ancien auquel de nombreux travaux font référence (citons l'étude [CML00]). Nous aborderons dans un premier chapitre une sélection de systèmes argumentatifs retenus pour les idées novatrices qui y ont été mises en œuvre. Après avoir exposé l'une des premières approches formelles du raisonnement révisable, nous verrons que la notion d'argument a été définie de plusieurs façons. Suivra une description de quelques relations de conflit entre arguments, permettant une première distinction entre les différents arguments d'un système dialectique. Finalement nous décrirons quelques travaux sur le concept de classes d'acceptabilité, autorisant une détermination globale de la qualité des arguments. Un argument sera jugé acceptable si aucun de ses contrariants ne l'est.

À la base de notre processus de génération d'arguments se trouve le concept d'attitudes. Nous les élaborerons à l'aide d'un outil mathématique particulier : la *X-inférence*, ou l'inférence des X-logiques ([SF96]). Les X-logiques — une extension non-monotone de la logique classique propositionnelle — n'avaient pas été utilisées auparavant au sein de systèmes argumentatifs, mais nous avons montré dans [Aub02], [AR05] et [AR06] qu'elles constituent une approche prometteuse pour la génération d'arguments : le fait d'être pourvues d'un ensemble de formules paramétrant le procédé d'inférence classique a des répercussions essentielles tant sur la définition d'un agent que sur son comportement vis-à-vis d'un argument. Le chapitre deux leur est dévolu et introduit la notion de *recevabilité* qui servira à ériger les attitudes.

Procédant étape par étape, nous donnons dans le chapitre trois la définition de nos agents, constitués de connaissances et d'un ensemble de formules interdites. Cet ensemble d'interdits est exploité par la particularité de la X-inférence. Nous introduisons finalement la notion d'attitude : un agent peut alors être *pour*, *neutre*, *contre* ou *perplexe* au sujet d'un ensemble de formules, selon la recevabilité de cet ensemble face aux connaissances et interdits de l'agent. Nous en profitons pour exposer la construction de ces attitudes, et démontrer que l'on ne peut en avoir davantage dans ce système. Par ailleurs, une étude poussée de l'évolution des attitudes est entreprise.

Tout ceci nous conduit dans un quatrième chapitre à introduire des opérateurs de confrontation, définis sur les attitudes : ils partitionnent un ensemble de formules selon les attitudes d'un agent à son égard. Par leur intermédiaire, un agent pourra ainsi accéder directement aux parties du support de l'argument qu'il analyse et envers lesquelles il est pour, contre ou autre. Il est montré dans un second temps que ces opérateurs de confrontation sont élaborables à partir d'opérateurs de base.

Le chapitre cinq est consacré à la notion de réponse. Les réponses seront les supports de nos arguments générés. Elles sont élaborées à partir des connaissances et des interdits des agents et sont toujours associées à un ensemble de formules. Plusieurs types de réponses sont distingués : des réponses pertinentes, des mensonges.

Au sein du sixième chapitre se trouve la définition de ce qu'est précisément un argument dans notre système. Des relations inter-arguments y sont également exposées, et discutées. Enfin la génération d'arguments devient effective et nous montrons, pour un agent et un support d'argument donnés, qu'à chaque ensemble produit par les opérateurs de confrontation correspond un argument attaquant ou défendant ce support, voire les deux dans le cas de la perplexité. Un exemple est présenté, illustrant l'analyse d'un argument, la génération d'un argument en retour, construit ou non sur des réponses pertinentes, et la génération d'un argument mensonger.

Une application de nos travaux est exposée dans un septième et dernier chapitre. Afin de réunir arguments et contre-arguments pour ou contre une thèse initiale, Besnard et Hunter construisent dans [BH01] des arbres argumentatifs au sein desquels s'imbriquent ces arguments, simulant de fait un débat. Le regroupement dans une structure argumentative de tous les arbres dont la racine statue sur un fait particulier permet alors de mesurer le crédit que l'on peut accorder à cette thèse. Ces arbres ont ceci de particulier qu'ils ne contiennent que des *contre-arguments conservatifs maximaux*, arguments retenus par Besnard et Hunter pour leur pertinence. Nous exposons alors comment générer de tels arguments à l'aide de notre formalisme.

Notations

Nous utilisons le langage de la logique propositionnelle, noté $\mathcal{L}$. Les lettres minuscules $a, b, c, \ldots, f, \ldots$ dénotent les formules du langage, tandis que les lettres majuscules $A, B, \ldots, E, \ldots$ dénotent des ensembles de formules. Les symboles $\top$ et $\bot$ codent respectivement la tautologie et la contradiction. Les connecteurs logiques sont notés $\neg, \wedge, \vee, \Rightarrow$ et $\Leftrightarrow$, pour, dans l'ordre, la négation, la conjonction, la disjonction, l'implication et l'équivalence. L'inférence classique est représentée par $\vdash$; par exemple, $E \vdash f$ signifie que la formule f est un théorème de l'ensemble E, et $E \nvdash \bot$ signifie que l'ensemble E est consistant...

L'ensemble vide est noté $\varnothing$. Un ensemble fini E de formules est interprété logiquement comme la conjonction de ses éléments. Nous introduisons la notation $\neg E$ comme un raccourci d'écriture désignant la négation de la conjonction des formules de E, et la notation $\bar{\cup} E$ qui représente l'union des négations des formules de E ; par exemple, si $E = \{e_1, \ldots, e_n\}$, alors $\neg E = \neg e_1 \vee \cdots \vee \neg e_n$ et $\bar{\cup} E = \{\neg e_1, \ldots, \neg e_n\}$. La cardinalité de E est calculée par $\mathsf{card}\,(E)$, et vaut donc n dans le court exemple précédent. L'ensemble des parties P de E — c'est-à-dire les sous-ensembles de E — est noté 2^E. Si E est un ensemble d'ensembles, nous désignons par $\mathsf{min}\,(E)$ l'ensemble des ensembles minimaux de E au sens de l'inclusion. Enfin, une base de connaissances est un ensemble fini et consistant de formules.

Les arguments sont représentés par des lettres grecques minuscules : $\alpha, \beta, \gamma, \ldots$ derrière lesquelles, notamment dans notre système, se cache un couple : $\alpha = \langle S, c\rangle$.

Précisons pour terminer que les agents sont appelés Φ ou Ψ, et sont constitués de deux ensembles de formules : $\Phi = [K, X]$.

Chapitre 1

Quelques systèmes argumentatifs

Nous allons présenter dans ce chapitre une sélection de systèmes argumentatifs retenus pour les idées novatrices qui y ont été mises en œuvre.

Chacun de nous s'est déjà retrouvé dans la situation de devoir convaincre un auditoire de la validité de son opinion, de son point de vue, que ce soit lors d'un entretien d'embauche ou lors de la présentation d'un projet par exemple. Nous avançons alors des raisons motivant notre point de vue. Et l'important n'est pas tant la véracité des motivations que le fait qu'on justifie notre opinion pour lui accorder plus de poids. Ces justifications vont constituer des arguments auxquels l'auditoire, étant rarement acquis à notre cause, opposera des contre-arguments. Ces derniers peuvent venir contrarier notre thèse ou l'une des raisons avancées pour la soutenir, et selon la pertinence de chacun il nous faudra éventuellement réviser notre jugement.

Le fait qu'un raisonnement puisse-t-être remis en question a donné lieu à la création par Rescher d'*assertions provisoires* (section 1.1). Nous assisterons ensuite à la formalisation du concept d'argument (section 1.2), liant raisons et conclusions. De nombreuses définitions ont été proposées, montrant s'il en était besoin qu'il s'agit là d'une notion clef du processus d'argumentation. Une première tentative de catégorisation d'arguments est alors avancée par Lin et Shoham au sein de *structures argumentatives* (section 1.3). Des relations de *conflit* inter-arguments on par suite été proposées et si leur diversité est sans pareil, deux types de conflits sont typiquement distingués (section 1.4). Cette notion de contrariété entre arguments est également une composante essentielle à tout système argumentatif puisqu'elle est un premier pas vers l'appréciation de la force ou la qualité d'un argument. L'idée de *préférence* entre arguments y est d'ailleurs incorporée, dans le but de mieux s'accomoder de l'incomplétude et de l'incertitude des connaissances. La section 1.5 traitera du statut ou de l'acceptabilité d'un argument, fonction de ses contre-arguments, des contre-arguments de ces derniers, et ainsi de suite. On parle alors de *classes d'acceptabilité*. Enfin la notion d'*acceptabilité graduelle* est le sujet de la section 1.6. L'idée est de dépasser les trois classes d'acceptablités généralement définies.

1.1 Des assertions provisoires

Les raisonnements de sens commun — comme ceux que nous effectuons au quotidien — s'accomodent d'informations incomplètes, incertaines, voire incohérentes. Les capacités de représentation et d'inférence de la logique classique sont dépassées par de tels processus de raisonnement. En effet les conclusions plausibles établies en cas d'informations incomplètes sont révisables puisqu'elles pourront être remises en cause à l'arrivée d'une nouvelle information. Les modèles d'argumentation sont alors un moyen pour essayer de saisir le raisonnement révisable, et l'on doit à Rescher l'une des premières approches formelles.

Dans son livre [Res77], Rescher décrit une discussion comme un échange de propositions faisant intervenir trois parties : le *proposant*, l'*opposant* et le *déterminateur*.

La racine de la discussion est la thèse initiale du proposant. À tour de rôle, le proposant et l'opposant vont effectuer un ou plusieurs *mouvements fondamentaux*.

Trois types de mouvements fondamentaux sont distingués :

- le mouvement *catégorique* : $!P$
  « P est vrai »,
- le mouvement *prudent* : $\dagger P$
  « P est compatible avec tout ce que l'on a vu »,
- le mouvement *conditionnel* : P/Q
  « Si Q est vrai alors P est vrai ».

Le déroulement d'une discussion est codifié. Le premier mouvement est toujours une assertion catégorique. Une réponse envisageable à un mouvement catégorique $!P$ peut être un mouvement prudent $\dagger\neg P$ ou conditionnel $\neg P/Q \wedge \dagger Q$. Une contre-attaque à une assertion prudente $\dagger P$ peut être un mouvement catégorique $!\neg P$ ou conditionnel $\neg P/Q \wedge !Q$. Les assertions provisoires sont considérées comme correctes et ne peuvent pas être discutées directement. On ne peut donc répondre $\neg P/Q$ à P/Q. Les attaques possibles d'une assertion provisoire $\neg P/Q$ sont soit une distinction *faible* $\neg P/(Q \wedge R) \wedge \dagger(Q \wedge R)$, soit une distinction *forte* $\neg P/(Q \wedge R) \wedge !(Q \wedge R)$. Le débat se déroule jusqu'à acceptation des arguments de l'un par l'autre. Si la limite de temps est dépassée alors le déterminer intervient et décide de l'issue de la discussion au regard des arguments avancés. Par ailleurs les mouvements catégoriques sont réservés au proposant, les mouvements prudents à l'opposant. Il est interdit de répéter deux fois la même chose et les concessions sont faites par défaut.

Exemple 1.1 ([Res77]). La figure 1.1 expose un débat au sein du système de Resher. Dans cet exemple, le proposant affirme que le kart est autorisé. L'opposant rétorque que si le kart est un véhicule alors il n'est pas autorisé. Et il lui semble qu'il s'agit d'un véhicule. Le proposant affirme que ce n'en est pas un...

	Proposant	**Opposant**
(1)	!autorisé (kart)	¬ autorisé (kart) / véhicule (kart) ∧ †véhicule (kart)
(2)	!¬véhicule (kart)	véhicule (kart) / motorisé (kart) ∧ †motorisé (kart)
(3)	!¬motorisé (kart)	motorisé (kart) / autopropulsé (kart) ∧ †autopropulsé (kart)
(4)	¬motorisé (kart) / (autopropulsé (kart) ∧ a-une-voile (kart)) ∧ !(autopropulsé (kart) ∧ a-une-voile (kart))	véhicule (kart) / a-des-roues (kart) ∧ †a-des-roues (kart)
(5)	¬véhicule (kart)/ (a-des-roues (kart) ∧ jouet (kart)) ∧ !(a-des-roues (kart) ∧ jouet (kart))	

FIG. 1.1 – Un débat dans le modèle de Rescher

$\diamondsuit$

La figure 1.2 représente un échange concernant la détermination de l'aptitude à voler de Tweety.

Entre autres choses, Rescher aborde aussi la notion de *poids d'un argument*. L'argument premier de la discussion, celui qui se trouve en être la racine, est intrinsèquement mis en avant par rapport aux autres. Un modèle argumentatif devrait être à même d'en rendre compte. De même les derniers

	Proposant	Opposant
(1)	$!\neg$vole	vole/oiseau $\wedge$ †oiseau
(2)	$!\neg$vole/(pingouin $\wedge$ oiseau)	†$\neg$pingouin
	$\wedge$!(pingouin $\wedge$ oiseau)	

FIG. 1.2 – Tweety dans le modèle de Rescher

arguments d'un débat sont eux aussi plus significatifs que les autres. En fait Rescher prétend qu'un argument reste *plausible* tant qu'il n'a pas été débouté : « Nous pensons que les oiseaux volent, Tweety est un oiseau, donc il *semblerait* que Tweety vole. » La proposition est acceptée par défaut.

Remarque 1.2 (Assertions provisoires et logique des défauts)**.** Avec ses assertions provisoires Rescher a jeté les premières bases de la théorie des défauts. C'est seulement quelques années plus tard que Reiter établira dans [Rei80] la *logique des défauts*, brique importante du raisonnement non-monotone.

Pour exemple, la croyance « généralement un oiseau vole », s'écrit avec le défaut suivant :

$$\text{Oiseau}(x) : \text{Peut-voler}(x) \,/\, \text{Peut-voler}(x)$$

ce qui peut s'interpreter comme : « Si je crois que Tweety est un oiseau et si rien dans mes croyances m'empêche de croire que Tweety peut voler, alors je crois que Tweety peut voler. » À partir de ce défaut et de : $\{\text{Oiseau}(\text{Tweety}), \forall x \ \text{Pingouin}(x) \Rightarrow \neg\text{Peut-voler}(x)\}$, on peut inférer *par défaut* Peut-voler(Tweety). Si l'on apprend Pingouin(Tweety), la conclusion $\neg$Peut-voler(Tweety) sera déduite classiquement et le défaut ne pourra plus s'appliquer, interdisant du même coup la conclusion contraire. $\diamondsuit$

Avec toujours comme objectif la manipulation d'informations incomplètes ou révisables, le concept d'assertion provisoire et de mouvement prudent de Rescher s'appellera *raison prima facie* chez [Pol87] en associant des poids aux règles, ou par exemple *règle révisable* chez [Vre91] avec l'ajout d'un second type d'inférence, quand le concept de défaut de Reiter n'a pas tout simplement été maintenu ([Lou87, LS89, SL92, PS96]).

1.2 La notion d'argument

On ne trouve pas à proprement parler de définition d'un argument au sein du modèle de Rescher. Il s'agit là d'une notion clef du processus d'argumentation et s'il n'existe toujours pas de consensus à son sujet, nous pouvons actuellement distinguer schématiquement trois architectures pour un argument : un arbre d'inférences, une séquence d'inférences ou simplement une paire support-conclusion.

Notons que Dung laisse dans [Dun95] la structure interne des arguments complètement non spécifiée pour se focaliser sur leurs interactions. Conjugué à une relation de conflit entre arguments des plus abstraites (*cf.* définition 1.11), Dung propose là un système argumentatif unifiant de nombreuses approches du raisonnement non-monotone.

1.2.1 Arbre d'inférences

Intuitivement un argument défini sous forme d'arbre est constitué de prémisses que l'on retrouve aux feuilles de cet arbre, et d'une conclusion placée au sommet de ce dernier. Les autres nœuds peuvent être qualifiés de conclusions intermédiaires.

Exemple 1.3. La figure 1.3 présente un argument contenant trois prémisses et une conclusion. Il implique une conclusion intermédiaire qui est que Tweety est un oiseau.

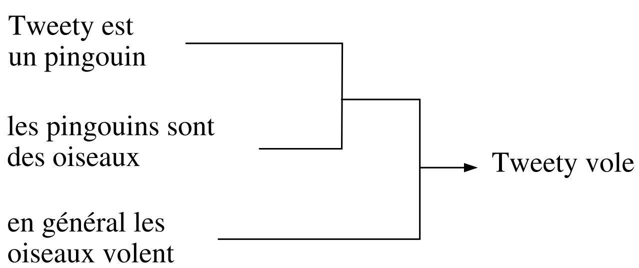


FIG. 1.3 – Argument sous forme d'arbre

$\diamondsuit$

Citons entre autres Vreeswijk et Lin et Shoham — nous abordons en section 1.3 leur système — qui utilisent cette approche ([Vre91, LS89]). Voici une formalisation que l'on doit à Vreeswijk :

Définition 1.4 (Argument, [Vre91])**.** *Soient $\mathcal{L}$ un langage et $\mathcal{R}$ un ensemble de règles d'inférences qui peuvent être strictes ($\Rightarrow$) ou révisables ($\rightarrow$). Un* *argument α possède des prémisses* ($\mathsf{prém}(\alpha)$)*, une conclusion* ($\mathsf{conc}(\alpha)$) *et des propositions* ($\mathsf{prop}(\alpha)$)*. α est soit :*

1. *une formule de $\mathcal{L}$, et dans ce cas,* $\mathsf{prém}(\alpha) = \{\alpha\}$, $\mathsf{conc}(\alpha) = \alpha$ *et* $\mathsf{prop}(\alpha) = \{\alpha\}$ *;*
2. *une formule de la forme $\alpha_1, \ldots, \alpha_n \Rightarrow f$, avec $\alpha_1, \ldots, \alpha_n$ une séquence finie d'arguments telle que* $f \notin \mathsf{prop}(\alpha_1) \cup \cdots \cup \mathsf{prop}(\alpha_n)$ *et* $\mathsf{conc}(\alpha_1), \ldots, \mathsf{conc}(\alpha_n) \Rightarrow f \in \mathcal{R}$*. Dans ce cas,* $\mathsf{prém}(\alpha) = \mathsf{prém}(\alpha_1) \cup \cdots \cup \mathsf{prém}(\alpha_n)$, $\mathsf{conc}(\alpha) = f$ *et* $\mathsf{prop}(\alpha) = \mathsf{prop}(\alpha_1) \cup \cdots \cup \mathsf{prop}(\alpha_n) \cup \{f\}$ *;*
3. *une formule de la forme $\alpha_1, \ldots, \alpha_n \rightarrow f$, avec $\alpha_1, \ldots, \alpha_n$ une séquence finie d'arguments telle que* $f \notin \mathsf{prop}(\alpha_1) \cup \cdots \cup \mathsf{prop}(\alpha_n)$ *et* $\mathsf{conc}(\alpha_1), \ldots, \mathsf{conc}(\alpha_n) \rightarrow f \in \mathcal{R}$*. Dans ce cas, prémisses, conclusion et propositions sont définies comme en 2.*

On remarquera que ces arguments satisfont un critère de minimalité en ce sens que leur conclusion ne doit pas faire partie des prémisses. Par ailleurs il n'est pas requis d'ordonner ces dernières.

1.2.2 Séquence d'inférences

Prakken et Sartor dans [PS96, PS97] ainsi que Pollock dans [Pol87] font état d'arguments constitués d'une séquence de règles chaînées entre elles.

Dans le système argumentatif développé par Prakken et Sartor, $\mathcal{F}$ est l'ensemble des faits — un ensemble de règles du premier ordre — et Δ représente les connaissances révisables sous forme de défauts de Reiter. Un argument y est défini comme suit :

Définition 1.5 (Argument, [PS96])**.** *Soit Γ une théorie des défauts $(\mathcal{F}, \Delta)$. Un argument de Γ est une séquence de formules distinctes du premier ordre ou de défauts $[f_1, \ldots, f_n]$ telle que pour tout f_i, soit :*
- *$f_i \in \Gamma$,*
- *il existe une règle d'inférence $g_1, \ldots, g_m / f_i$ à base de formules du premier ordre ou de défauts telle que $g_1, \ldots, g_m \in \{f_1, \ldots, f_{i-1}\}$.*

Pour tout argument α :
- *$f \in \alpha$ est une* conclusion *de α ssi f est une formule du premier ordre,*
- *$f \in \alpha$ est une* hypothèse *de α ssi f est une hypothèse de l'un des défauts de α.*

Outre le fait que de tels arguments peuvent avoir plusieurs conclusions, ils ne sont pas supposés consistants. Par ailleurs cette définition impose que leurs formules soient ordonnées.

1.2.3 Paire prémisses-conclusion

Une autre manière de définir un argument est de le considérer comme une paire prémisses-conclusion où les prémisses sont une collection non ordonnée de formules qui infère classiquement une unique conclusion.

Dans le cadre du traitement de l'incohérence au sein de bases de connaissances, Simari et Loui ([SL92], voir section 1.4) et Elvang-Gøransson, Krause et Fox ([EGKF93]) ont développé des systèmes d'argumentation utilisant cette approche par paire en ajoutant des critères de minimalité et de consistance aux arguments. Citons Amgoud et Cayrol ([AC98]) ainsi que Besnard et Hunter ([BH01]) qui définissent également un argument de la sorte.

Définition 1.6 (Argument, [BH01])**.** *Soit* Δ *un ensemble fini et éventuellement inconsistant de formules représentant les connaissances et croyances du système. Un* argument *est un couple* $\langle S, c\rangle$ *tel que :*

1. $S \nvdash \bot$
2. $S \vdash c$
3. S *est un sous-ensemble minimal de* Δ *satisfiant 2.*

S *est appelé le* support *de l'argument et* c *sa* conclusion*.*

Dans nos travaux nous avons également opté pour des arguments sous forme de paires, non seulement parce que cette implémentation semble la plus simple des trois, mais aussi parce que nous comparons nos résultats avec une partie de ceux de Besnard et Hunter. Néanmoins, si les définitions exposées dans les travaux que nous venons de citer se ressemblent beaucoup, la nôtre (définition 6.1) se présente autrement du fait de l'incorporation de notions d'agent et de réponse.

1.3 Une structure argumentative

Lin et Shoham présentent dans [LS89] un système argumentatif entièrement basé sur des règles d'inférence définies sur un langage logique non spécifié — il est uniquement supposé contenir un symbole de négation. Ils montrent comment une logique monotone ou non peut être reformulée comme un système de construction d'arguments.

On peut considérer leur système comme le premier travail sur l'argumentation largement accepté par la communauté en Intelligence artificielle.

Il faut distinguer dans leur système trois types de règle d'inférence (où A_i et B sont des formules bien formées dans un langage quelconque et où $n > 0$) :

- *un fait de base :* A
  cela représente les connaissances explicites, les prémisses,
- *une règle monotone :* $A_1, \ldots, A_n \vdash B$
  elles décrivent les connaissances déductives,
- *une règle non-monotone :* $A_1, \ldots, A_n \mathrel{|\!\!\sim} B$
  pour décrire le sens commun.

En les enchaînant dans des arbres, ces trois genres d'inférence sont utilisés pour construire des arguments, dans lesquels une formule ne doit y apparaître qu'une fois au plus. Un argument supporte une formule f si et seulement si le sommet de l'arbre de l'argument est f.

Un concept important introduit par Lin et Shoham est celui de *structure argumentative* :

Définition 1.7 ([LS89]). *Un ensemble T d'arguments est une* structure d'arguments *si et seulement si :*

- *les faits de base sont dans T,*
- *T est clos : $\forall a \in T$, si a' est un sous-arbre de a alors $a' \in T$,*
- *T est clos par rapport à la monotonie : si a est construit à partir de $a_1, \ldots, a_n \in T$ par une règle monotone, alors $a \in T$,*
- *T est consistant.*

Exemple 1.8 (Tweety chez Lin et Shoham). Considérons le jeu de règles suivant ([LS89]) :

r_1 : true

r_2 : pingouin(t)

r_3 : pingouin(t) $\vdash$ oiseau(t)

r_4 : pingouin(t) $\wedge$ $\neg$anormal(pingouin(t)) $\vdash$ $\neg$vole(t)

r_5 : oiseau(t) $\wedge$ $\neg$anormal(oiseau(t)) $\vdash$ vole(t)

r_6 : pingouin(t) $\vdash$ anormal(oiseau(t))

r_7 : true $\mathrel{|\!\!\sim}$ $\neg$anormal(pingouin(t))

r_8 : true $\mathrel{|\!\!\sim}$ $\neg$anormal(oiseau(t))

Ce jeu de règles nous permet d'élaborer les arguments suivants :

a_1 : true

a_2 : pingouin(t)

$a_3 : a_1 \overset{r_7}{\mid\!\sim} \neg$anormal(pingouin$(t)$)

$a_4 : a_1 \overset{r_8}{\mid\!\sim} \neg$anormal(oiseau$(t)$)

$a_5 : a_2 \overset{r_3}{\vdash}$ oiseau(t)

$a_6 : a_2 \overset{r_6}{\vdash}$ anormal(oiseau(t))

$a_7 : a_3,\ a_2 \overset{r_4}{\vdash} \neg$vole$(t)$

$a_8 : a_4,\ a_5 \overset{r_5}{\vdash}$ vole(t)

Nous pouvons alors construire les structures d'arguments suivantes :

$T_1 = \{a_1, a_2, a_5, a_6\}$

$T_2 = \{a_1, a_2, a_6, a_3, a_7\}$

$\diamondsuit$

Ce système ne comporte pas de critère de comparaison entre arguments conflictuels, aspect que nous étudions dans la prochaine section. La certitude d'une proposition ne peut donc y être évaluée, mais Lin et Shoham parviennent à répartir des arguments en structures argumentatives. Et si ces structures ne sont pas appréciables l'une par rapport à l'autre, elles sont vraisemblablement les précurseurs des classes d'acceptabilité qui permettront de déterminer entre autre si un argument doit être retenu ou non (*cf.* section 1.5).

À noter que Vreeswijk présentera dans [Vre92] un système rappelant l'approche de Lin et Shoham mais en ajoutant justement des relations de conflit et de préférence entre arguments.

1.4 Conflits entre arguments

Il existe une différence majeure entre preuves et arguments. Tandis que les preuves justifient leurs conclusions, un argument peut être réfuté par des *contre-arguments*. La notion de contrariété entre arguments est une composante essentielle à tout système argumentatif puisqu'elle est un premier pas vers l'appréciation de la *force* ou la *qualité* d'un argument, comme nous le verrons dans la section suivante.

1.4.1 Une illustration

Simari et Loui ont proposé un système ([SL92]) combinant plusieurs travaux que nous allons détailler : les idées de Pollock sur l'interaction des arguments (*cf.* définition 1.19), le travail sur la spécificité des arguments de Poole ([Poo88]) et les idées de Loui sur les défauts ([Lou87]). Leur principal souci a été de caractériser les conditions sous lesquelles une structure d'argument est préférée à une autre, ce qui, comme nous l'avons souligné, faisait cruellement défaut au modèle de Lin et Shoham (section précédente).

Les connaissances d'un agent sont modélisées par un couple (K, Δ) où l'ensemble K représente les connaisances sûres elles-mêmes partagées entre faits sûrs — K_C, pour *contingent knowledge* — et règles sûres — K_N, pour *necessary knowledge* —, et Δ un ensemble fini de règles révisables de type : $p \rightarrowtail q$, signifiant « p est une raison pour q ».

Loui est à l'origine de ces règles annulables que l'on peut voir comme des défauts à deux emplacements ([Lou87]). Ce fait distingue le système de Simari et Loui d'avec celui de Prakken et Sartor manipulant des défauts de type Reiter — systèmes qui sont par ailleurs relativement similaires : voir [PS96] et notamment la version de [Pra97]. Il en découle que Prakken et Sartor sont à même de différencier un autre type d'attaque entre arguments, à savoir lorsque la conclusion d'un argument attaque les hypothèses faites au sein d'un défaut (la partie B) de l'autre l'argument.

Simari et Loui conçoivent leurs arguments sous forme de paires $\langle$*prémisses*, *conclusion*$\rangle$ (*cf.* section 1.2.3) : $\langle S', c'\rangle$ est un *sous-argument* de $\langle S, c\rangle$ si S' est inclus dans S.

En exploitant les travaux de Poole sur la spécificité des arguments (voir [Poo88]) — un argument est plus spécifique qu'un autre s'il est plus contraignant : exemple 1.10 —, Simari et Loui avancent différents types de conflits entre arguments.

Définition 1.9 (Conflits entre arguments, [SL92]). *Soient* $\langle A_1, h_1\rangle$ *et* $\langle A_2, h_2\rangle$ *deux arguments :*

- $\langle A_1, h_1\rangle$ *et* $\langle A_2, h_2\rangle$ *sont en* désaccord *ssi* $K \cup \{h_1, h_2\} \vdash \perp$*,*
- $\langle A_1, h_1\rangle$ contredit $\langle A_2, h_2\rangle$ *en* h *ssi il existe un sous-argument* $\langle A, h\rangle$ *de* $\langle A_2, h_2\rangle$ *tel que* $\langle A_1, h_1\rangle$ *et* $\langle A, h\rangle$ *sont en désaccord,*
- $\langle A_1, h_1\rangle$ bat $\langle A_2, h_2\rangle$ *ssi il existe un sous-argument* $\langle A, h\rangle$ *de* $\langle A_2, h_2\rangle$ *tel que* $\langle A_1, h_1\rangle$ *contredit* $\langle A, h\rangle$ *et* $\langle A_1, h_1\rangle$ *est soit strictement plus spécifique que* $\langle A, h\rangle$ *(*proper defeaters*) soit sans rapport de spécifité avec* $\langle A, h\rangle$ *(*blocking defeaters*).*

Exemple 1.10 ([CML00]). Soit v une voiture. Posons les connaissances suivantes :

$K_C = \{$
propulsion(v),
italienne(v),
faible-consommation(v),
n'est-plus-produite(v),
a-des-pièces-détachées$(v)\}$,
$K_N = \{$italienne$(x) \Rightarrow$ européenne$(x)\}$,
$\Delta = \{$
fiable$(x) \rightarrowtail$ recommandable(x),
fiable$(x) \wedge$ n'est-plus-produite$(x) \rightarrowtail \neg$recommandable(x),
européenne$(x) \wedge$ a-des-pièces-détachées$(x) \rightarrowtail \neg$fiable(x),
propulsion$(x) \rightarrowtail$ système-arrière(x),
faible-consommation$(x) \wedge$ système-arrière$(x) \rightarrowtail \neg$fiable(x),
propulsion$(x) \rightarrowtail$ problèmes-de-refroidissement(x),
problèmes-de-refroidissement$(x) \rightarrowtail \neg$fiable$(x)\}$.

Parmi les arguments que l'on peut construire à partir de ces connaissances citons A_1 pour recommandable(v), A_2 pour $\neg$recommandable(v), A_3 pour $\neg$fiable(v) et A_4 pour fiable(v), où :

$A_1 = \{$
européenne$(v) \wedge$ a-des-pièces-détachées$(v) \rightarrowtail$ fiable(v),
fiable$(v) \rightarrowtail$ recommandable$(v)\}$,
$A_2 = \{$
européenne$(v) \wedge$ a-des-pièces-détachées$(v) \rightarrowtail$ fiable(v),
fiable$(v) \wedge$ n'est-plus-produite$(v) \rightarrowtail \neg$recommandable$(v)\}$,
$A_3 = \{$
propulsion$(v) \rightarrowtail$ problèmes-de-refroidissement(v),
problèmes-de-refroidissement$(v) \rightarrowtail \neg$fiable$(v)\}$,
$A_4 = \{$
propulsion$(v) \rightarrowtail$ système-arrière(v),
faible-consommation$(v) \wedge$ système-arrière$(v) \rightarrowtail$ fiable$(v)\}$.

Il en résulte les conflits inter-arguments suivants :
- $\langle A_3, \neg\text{fiable}(v)\rangle$ et $\langle A_4, \text{fiable}(v)\rangle$ sont en désaccord,
- $\langle A_3, \neg\text{fiable}(v)\rangle$ contredit $\langle A_1, \text{recommandable}(v)\rangle$,
- $\langle A_2, \neg\text{recommandable}(v)\rangle$ bat $\langle A_1, \text{recommandable}(v)\rangle$ et est un *proper defeater* pour ce dernier,
- enfin $\langle A_3, \neg\text{fiable}(v)\rangle$ bat $\langle A_2, \neg\text{recommandable}(v)\rangle$ et est un *blocking defeater* pour ce dernier.

$\diamondsuit$

1.4.2 De nombreux types de contrariété

Bien d'autres relations de contrariété ont été définies dans la littérature (mentionnons les études de [CML00] et [PV02]). Mais typiquement, et même si les terminologies varient, depuis les travaux de Pollock ([Pol70]) deux types de conflits sont distingués. Il y a d'une part la *réfutation* d'un argument qui survient lorsque les conclusions d'arguments sont contradictoires : par exemple « Tweety vole parce que c'est un oiseau. » et « Tweety ne vole pas, parce que c'est un pingouin. » sont deux arguments qui se réfutent l'un l'autre. Cette relation est symétrique, contrairement à la seconde forme de conflit que nous nommons *opposition* (*cf.* définition 6.3 sur nos relations inter-arguments) et qui apparaît lorsqu'un argument réfute une partie du support ou prémisses d'un autre argument : tout argument de conclusion « Tweety est un pingouin. » s'opposera à « Tweety vole parce que c'est un oiseau et il n'est pas prouvé que Tweety est un pingouin. »

La réfutation et l'opposition sont respectivements nommés *désaccord* et *contradiction* dans la définition 1.9 de Simari et Loui, *reductio ad absurdum attack* et *ground attack* dans [Dun93a, Dun93b], *réfutation* et *attaque* chez [AC98] ou encore *rebuttal* et *undercut* par [BH01]. À part, dans [Dun95], Dung propose ce qui est probablement la relation d'attaque entre arguments la plus abstraite qui soit :

Définition 1.11 ([Dun95])**.** *Un système d'argumentation est un couple noté* $\langle \mathcal{A}rg, \mathcal{R} \rangle$ *avec* $\mathcal{A}rg$ *un ensemble d'arguments et* $\mathcal{R}$ *une relation binaire représentant la relation de contrariété entre les arguments, c'est-à-dire* $\mathcal{R} \subseteq \mathcal{A}rg \times \mathcal{A}rg$*.* $(\alpha, \beta) \in \mathcal{R}$ *ou, d'une manière équivalente,* $\alpha \mathcal{R} \beta$ *veut dire que l'argument* α *contrarie l'argument* β*.*

Soulignons le fait que Besnard et Hunter définissent dans [BH01] une nouvelle relation de conflit entre arguments appelée *canonical undercutting* qu'ils établissent comme étant la seule dont ils ont besoin de tenir compte, la réfutation et l'opposition pouvant s'y ramener. Ceci leur permet d'établir une sélection drastique des arguments que nous essayerons de reproduire (section 7).

1.4.3 Des préférences pour apprécier l'incertitude

Dans le système de Simari et Loui comme dans d'autres ([Gro91, Bre94, CLS94, GS03]), des relations de *préférences* entre arguments sont mises en œuvre afin d'obtenir un traitement plus sophistiqué des conflits et de l'incertitude des connaissances. Ici il s'agit de la notion de spécificité, avantageant les arguments dont les prémisses sont les plus nombreuses, mais d'autres critères peuvent être pris en compte. Par exemple il est possible de préférer les arguments les plus récents, privilégiant les nouvelles informations, ou bien alors, à l'image de [BDP93, EGKF93] ou encore Amgoud et Cayrol ([AC98]), imposer des priorités explicites sur les croyances, en les « stratifiant » — par exemple en décomposant l'ensemble de connaissances en sous-ensembles ordonnés dont chacun est préféré au précédent. Par suite les préférences amènent à l'émergence d'arguments se défendant seul face à un contre-argument. On parle alors de *défense indivduelle*.

Définition 1.12 (Défense individuelle, [AC98])**.** *Soient α et β deux arguments. α se* défend seul *contre β si et seulement si β s'oppose à α et α est préféré à β.*

Besnard et Hunter proposent une relation de préférence bien singulière (*cf.* définition 1.24) : nommée *conservativité*, elle avantage les arguments concis, ceux qui vont à l'essentiel. Cette notion leur permet de mesurer le crédit que l'on peut apporter à une thèse en ne tenant compte que des arguments les plus pertinents. Mais attention, car contrairement aux autres relations présentées plus haut, celle-ci n'a cure du départage des arguments conflictuels. Elle fournit un moyen de sélectionner le candidat le plus pertinent au grade de contre-argument.

1.4.4 Sur une approche éristique

Il est à noter que les relations entre arguments ne sont vues que sous l'aspect du conflit. On ne trouve pas de relations de défense ou de soutien directement d'argument à argument. Or il nous paraît sensé d'accorder plus de valeur à un argument soutenant une thèse répandue, entendre par là une conclusion partagée par nombre d'arguments. Et même si à défaut d'une approbation de cette conclusion il est d'autres arguments appuyant les prémisses de ce premier, alors le mettre en avant par rapport à des arguments « isolés » ne semble pas choquant. De cette idée découleront des relations d'approbation et de défense inter-arguments qui seront exploitées lors de la génération d'arguments (voir section 6.1 dédiée aux relations inter-arguments).

1.5 Statut d'un argument et classes d'acceptabilité

Les relations inter-arguments ne fournissent pas d'évaluation les concernant. L'enjeu des systèmes argumentatifs est pourtant de déterminer la

« qualité » de ceux-ci. On cherche à connaître les arguments *justifiés* et le statut d'un argument dépend de l'ensemble de ses interactions avec les autres. Cette considération a conduit au concept d'*acceptabilité*. Intuitivement, si un argument est acceptable ou justifié, alors aucun de ses contrariants ne doit l'être.

Exemple 1.13 (Rétablissement)**.** Soient trois arguments α, β et γ tels que β attaque α et γ attaque β (au sens de Dung, définition 1.11).

$$\alpha \longleftarrow \beta \longleftarrow \gamma$$

Plus concrètement :
- $\alpha =$ « Tweety vole parce que c'est un oiseau. »,
- $\beta =$ « Tweety ne vole pas, parce que c'est un pingouin. »,
- $\gamma =$ « Le fait que Tweety est un pingouin n'est pas fiable. ».

L'argument α est attaqué par l'argument β, lui-même attaqué l'argument γ. On choisira de juger α acceptable car son contrariant β est affaibli par γ. L'argument α est ainsi *rétabli* par γ, ou γ *défend* α. $\diamondsuit$

Une formulation de la justification des arguments peut se faire en termes de *classes d'acceptabilité*. Il en est de deux sortes.

1.5.1 L'acceptabilité individuelle

L'acceptabilité individuelle consiste à attribuer un niveau d'acceptabilité à chaque argument en fonction de l'existence de contrariants directs. Les arguments ayant le même niveau sont ensuite regroupés en classes ([EGKF93, EGH95]) : celles des arguments acceptés, celle des arguments rejetés — contrariés par un argument accepté — et celle des arguments défendables ou en suspens — contenant ceux ni acceptés ni rejetés.

Définition 1.14 (adapté de [EGKF93])**.** *La classe d'acceptabilité du système d'argumentation* $\langle \mathcal{A}rg, \mathcal{R} \rangle$ *est l'ensemble* $\{\alpha \in \mathcal{A}rg \mid \nexists \beta \in \mathcal{A}rg$*, tel que* $\beta \mathcal{R} \alpha\}$*. Chaque relation de contrariété définit une classse.*

La notion d'acceptabilité utilisée ici est très *prudente*. Elle n'accepte qu'un nombre très restreint d'arguments, à savoir ceux en dehors de tout conflit. Dans l'exemple 1.13 sur le rétablissement, seul l'argument γ est accepté...

1.5.2 L'acceptabilité conjointe

L'acceptabilité conjointe, principalement introduite dans [Dun93b], permet de caractériser des ensembles d'arguments conjointement acceptables. De tels ensembles sont appelés des *extensions*.

Cette forme d'acceptabilité est réalisée en recourant à une théorie de point fixe ([Dun95, PS97]) :

Définition 1.15 ([Dun95])**.** *Un argument α est* acceptable *pour un ensemble S d'arguments si et seulement si pour tout argument β tel que $\beta\mathcal{R}\alpha$ il existe un argument γ de S tel que $\gamma\mathcal{R}\beta$.*

Définition 1.16 ([Dun95])**.** *Soit $\langle \mathcal{A}rg, \mathcal{R}\rangle$ un système d'argumentation et $\mathcal{F}$ une fonction définie comme suit :*

$$\begin{array}{llll} \mathcal{F}: & 2^{\mathcal{A}rg} & \longrightarrow & 2^{\mathcal{A}rg} \\ & S & \longmapsto & \mathcal{F}(S) = \{\alpha \in \mathcal{A}rg \mid \alpha \textit{ est acceptable pour } S\} \end{array}$$

Dung définit ensuite deux sémantiques : la sémantique *complète* et la sémantique *basique*. Chacune génère ce que l'on appelle des *extensions*, c'est-à-dire des ensembles *points fixes*[1] de l'application $\mathcal{F}$. L'idée de la première est de capturer tous les arguments que l'on peut défendre. Elle correspond à une sémantique *crédule* et ses extensions sont chacun des points fixes de $\mathcal{F}$. La seconde est une sémantique *sceptique*. Elle possède pour seule extension le plus petit point fixe de $\mathcal{F}$, ou dit autrement, la plus petite extension de la sémantique complète.

Dans le premier cas un argument est accepté s'il appartient à au moins une extension, tandis que dans l'autre il lui faut être présent dans chaque extension.

Par ailleurs Dung a montré dans cet article que si $\mathcal{R}$ est fini alors $\mathcal{F}$ est continue et donc le plus petit point fixe de $\mathcal{F}$ peut être calculé en appliquant itérativement la fonction $\mathcal{F}$ à l'ensemble vide.

Définition 1.17 (Classes d'acceptabilité, [Dun95, PS97, PV02])**.**

- *Un argument est* justifié *si et seulement si il appartient au plus petit point fixe de $\mathcal{F}$.*
- *Un argument est* rejeté *si et seulement si il est non justifié et est contrarié par un argument justifié.*
- *Un argument est* défendable *si et seulement si il n'est ni justifié ni rejeté.*

[1] Un *point fixe* de $\mathcal{F}$ est un élément S de $2^{\mathcal{A}rg}$ tel que $\mathcal{F}(S) = S$.

Revenons à l'exemple 1.13 sur le rétablissement d'un argument pour en calculer les extensions :
- $\mathcal{F}(\varnothing) = \{\gamma\}$,
- $\mathcal{F}(\{\gamma\}) = \{\alpha, \gamma\}$,
- $\mathcal{F}(\{\alpha, \gamma\}) = \{\alpha, \gamma\}$.

Ainsi α et γ sont des arguments justifiés, et comme $\gamma \mathcal{R} \beta$ alors β est rejeté.

Dans des systèmes plus complexes, en particulier ceux où des arguments s'attaquent mutuellement, il peut exister plusieurs extensions. Le grand « classique » suivant en est un exemple :

Exemple 1.18 (*Nixon Diamond*)**.** Soient deux arguments α et β tels que α attaque β et β attaque α.

$$\alpha \rightleftarrows \beta$$

Par exemple :
- $\alpha =$ « Nixon était pacifiste parce qu'il était un quaker. »,
- $\beta =$ « Nixon n'était pas pacifiste, parce qu'il était un républicain. ».

$\diamondsuit$

On trouve trois points fixes dans cet exemple : $\varnothing$, $\{\alpha\}$ et $\{\beta\}$. Il en ressort que α et β sont des arguments défendables.

Avec ses arguments *in* et *out* — repris par Simari et Loui ([SL92]) et Vreeswijk ([Vre92]) —, Pollock capture également ces extensions :

Définition 1.19 (Statut des arguments, [Pol92])**.**

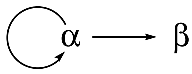
1. *Un argument est* in au niveau 0 *ssi il ne s'attaque pas.*
2. *Un argument est* in au niveau $n + 1$ *ssi il est* in *au niveau 0 et qu'il n'est battu par aucun argument* in *au niveau n.*
3. *Un argument est* justifié *ssi il existe un m tel que pour tout $n > m$, l'argument est* in *au niveau n.*

Appliquée à l'exemple 1.13 cette définition produit les faits suivants :
- au niveau 0, α, β et γ sont *in* ;
- seul γ reste *in* au niveau 1 ;
- au niveau 2 l'argument α rejoint γ ;
- les niveaux 3 et supérieurs sont identiques au niveau 2.

Ainsi l'on dira que α et γ sont justifiés, et β rejeté. Nous obtenons également des résultats identiques à ceux de Dung sur l'exemple de Nixon (exemple 1.18).

On notera que la définition 1.19 écarte les arguments qui s'attaquent eux-mêmes — appelés *self-defeating* dans la littérature — et que l'on juge irrecevables. Sans cette précaution l'argument β de l'exemple suivant aurait été injustement écarté — au lieu d'être justifié — suite à l'attaque qu'il subit de l'argument α, quand bien même ce dernier apparaît irrationnel vu qu'il s'attaque lui-même.

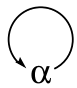
Exemple 1.20. Soient deux arguments α et β tels que α attaque simultanément β et lui-même.



♢

Une illustration d'argument s'auto-attaquant peut-être trouvé en l'exemple bien connu du paradoxe du menteur, décliné en d'innombrables versions. Une parmi d'autres :

Exemple 1.21 (Paradoxe du menteur)**.** Soit α un argument s'attaquant lui-même.



Plus précisément :

– α = « Frère Jean-Baptiste, un moine, dit qu'il est un menteur. Puisque les moines sont connus pour toujours dire la vérité, ce qu'il dit est vrai. Il est donc un menteur. »

♢

1.5.3 Arbres dialectiques

Dans [SCG94], Simari et Loui font évoluer leur système argumentatif. Au lieu de parler de niveau d'argument (*cf.* définition 1.19) ils introduisent la notion d'*arbre dialectique*. La thèse initiale de la discussion est alors la racine de l'arbre et à chaque nœud correspond un argument. Un étiquetage consiste à attribuer à chaque nœud de l'arbre la valeurs U (*undefeated*) ou D (*defeated*). Les feuilles de l'arbre sont toutes de type U. Les autres nœuds sont de type D si et seulement si ils ont au moins un fils de type U, ou de type U si et seulement si chacun de leurs fils sont de type D. Certaines contraintes doivent être satisfaites : pas d'attaque cyclique, pas d'arguments contradictoires, l'ensemble des niveaux pairs (respectivement impairs) doit être consistant.

Exemple 1.22. En reprenant l'exemple 1.10 sur les voitures il est possible de construire un arbre dialectique, représenté par la figure 1.4, statuant sur l'argument $\langle A_1, \text{recommandable}(v)\rangle$. Le résultat de son étiquetage conduit à la conclusion que cet argument n'est pas justifié.

$$
\begin{array}{ccc}
 & D & \\
 & \langle A_1, \text{recommandable}(v)\rangle & \\
\nearrow & & \nwarrow \\
D & & U \\
\langle A_3, \neg\text{fiable}(v)\rangle & & \langle A_2, \neg\text{recommandable}(v)\rangle \\
\uparrow & & \\
U & & \\
\langle A_4, \text{fiable}(v)\rangle & &
\end{array}
$$

FIG. 1.4 – Arbre dialectique

$\diamondsuit$

Chesñevar analysera dans [Che96] des stratégies pour accélerer la construction et l'étiquetage de tels arbres. García rebondira sur les travaux de Simari et Loui pour définir un langage de programmation logique révisable (DLP, [Gar97]) capable de capturer des aspects du raisonnement de sens commun, permettant la représentation et l'utilisation d'informations incomplètes, voire inconsistantes.

Ces arbres dialectiques sont en quelque sorte une retranscription de la définition 1.19 portant sur la justification par niveau des arguments et n'affinent pas les classes d'acceptabilité déjà proposées.

1.6 Vers une acceptabilité graduelle

Cayrol et Lagasquie-Schiex présentent dans [CLS05] un raffinage des classes d'acceptabilité de Dung (définition 1.17). Les auteurs prennent en compte à la fois le statut de l'argument par rapport à ses contrariants mais aussi une évaluation graduelle des arguments dans la notion d'acceptabilité conjointe. Cette évaluation induit un pré-ordre total sur l'ensemble des arguments et est fonction, pour un argument donné, de l'évaluation de tous les attaquants directs de ce dernier. L'évaluation est définie comme étant maximale pour un argument sans attaquant et non maximale pour un argument attaqué et non défendu.

Une toute autre forme d'acceptabilité graduelle est proposée par Besnard et Hunter.

1.6.1 Catégoriseurs et accumulateurs

Dans [BH01], Besnard et Hunter s'attachent à dépasser les classes d'acceptabilité définies précedemment pour déterminer une *acceptation graduelle*. Un argument y est présenté comme une paire prémisses-conclusion (définition 1.6). Les auteurs montrent que tout type de relation d'attaque entre arguments peut se ramener à une seule : de chaque argument en conflit avec un argument α peut ainsi être déduit un *undercut* pour α, que nous traduirons par *contre-argument*.

Définition 1.23 (Contre-argument, [BH01])**.** *Un* contre-argument *pour un argument* $\langle A, a\rangle$ *est un argument* $\langle B, \neg(a_1 \wedge \cdots \wedge a_n)\rangle$*, où* $\{a_1, \ldots, a_n\} \subseteq A$*.*

Un contre-argument pour un argument α est donc un argument qui s'oppose à α (*cf.* section 1.4.2 traitant des différents types de contrariété). Besnard et Hunter introduisent alors une notion bien singulière : nommée *conservativité*, elle doit leur permettre de sélectionner les arguments les plus pertinents parmi les contre-arguments afin d'aller à l'essentiel lors de la construction et de l'évaluation d'arbres à venir.

Définition 1.24 (Conservativité, [BH01])**.** *Un argument* $\langle A, a\rangle$ *est* plus conservatif *qu'un argument* $\langle B, b\rangle$ *ssi* $A \subseteq B$ *et* $b \vdash a$*.*

De la fusion des idées de ces deux définitions naît la notion de *contre-argument conservatif maximal* (Ccm) :

Définition 1.25 (Ccm, [BH01])**.** $\langle B, b\rangle$ *est un* contre-argument conservatif maximal *pour* $\langle A, a\rangle$ *ssi* $\langle B, b\rangle$ *est un contre-argument pour* $\langle A, a\rangle$ *tel qu'aucun autre contre-argument pour* $\langle A, a\rangle$ *est strictement plus conservatif que* $\langle B, b\rangle$*.*

Ainsi il est possible de réduire l'ensemble des contre-arguments aux plus pertinents. On ne retiendra pas un contre-arguments si l'on en a une version plus condensée.

Propriété 1.26 ([BH01])**.** *Si* $\langle B, \neg(a_1 \wedge \cdots \wedge a_n)\rangle$ *est un contre-argument conservatif maximal pour l'argument* $\langle A, a\rangle$*, alors* $A = \{a_1, \ldots, a_n\}$*.*

Par la suite on utilisera le symbole $\diamond$ pour noter la conséquence d'un CCM lorsqu'il n'y aura pas d'ambiguïté.

Définition 1.27 (Arbre argumentatif, [BH01])**.** *Un* arbre argumentatif *en faveur de f est un arbre dont les nœuds sont des arguments tels que :*

1. *la racine est un argument en faveur de f, et l'ancêtre de tout autre nœud ;*
2. *aucun nœud $\langle A, a\rangle$ ayant comme ancêtres $\langle A_1, a_1\rangle, \ldots, \langle A_n, a_n\rangle$ n'a $A \subseteq A_1 \cup \cdots \cup A_n$;*
3. *les fils d'un nœud sont tous les* CCM *de ce nœud qui vérifient 2.*

Un *arbre argumentatif* décrit la façon dont les arguments s'enchaînent au fil de la discussion. Ces arbres ont des propriétés fortes. La condition *2.* oblige chaque argument à apporter des éléments nouveaux, et donc empêche la discussion de boucler. La condition *3.* permet d'écarter les arguments réarrangés[2]. On montre par la définition des CCM (définition 1.25) associée aux conditions *2.* et *3.* que ces arbres sont finis aussi bien en profondeur qu'en largeur.

Exemple 1.28 ([BH01])**.** Soit $\Delta = \{a \Leftrightarrow \neg d, b, b \Rightarrow a, c \wedge \neg b, \neg c, d, \neg d\}$, l'ensemble de connaissances commun. À partir de cet ensemble il est possible de construire l'arbre proposé sur la figure 1.5, en faveur de la formule a.

$$\begin{tikzcd}
& \langle\{b \Rightarrow a, b\}, a\rangle & \\
\langle\{c \wedge \neg b\}, \diamond\rangle \arrow[ur] & & \langle\{d, a \Leftrightarrow \neg d\}, \diamond\rangle \arrow[ul] \\
\langle\{\neg c\}, \diamond\rangle \arrow[u] & & \langle\{\neg d\}, \diamond\rangle \arrow[u]
\end{tikzcd}$$

FIG. 1.5 – Arbre argumentatif en faveur de a

$\diamondsuit$

Afin de mesurer le crédit que l'on peut accorder à un fait — ici une formule — Besnard et Hunter commencent par regrouper tous les arbres argumentatifs en faveur et contre celui-ci au sein d'une *structure argumentative* :

Définition 1.29 (Structure argumentative, [BH01])**.** *Une* structure argumentative *pour une formule f est une paire d'ensembles $\langle \mathcal{P}, \mathcal{C}\rangle$ où $\mathcal{P}$ est*

[2]Deux contre-arguments $\langle C \cup A, \neg b\rangle$ et $\langle C \cup B, \neg a\rangle$ sont dit réarrangés ssi a est $a_1 \wedge \cdots \wedge a_n$ tel que $A = \{a_1, \ldots, a_n\}$ et b est $b_1 \wedge \cdots \wedge b_n$ tel que $B = \{b_1, \ldots, b_n\}$.

l'ensemble des arbres argumentatifs pour f et $\mathcal{C}$ l'ensemble des arbres argumentatifs pour $\neg f$.

Exemple 1.30 ([BH01])**.** Soit $\Delta = \{a \vee b, a \Rightarrow c, \neg c, \neg b, d \Leftrightarrow b\}$. La structure argumentative pour $a \vee \neg d$ est représentée par la figure 1.6.

$$\mathcal{P}\left\{\begin{array}{cc} \langle\{a \vee b, \neg b\}, a \vee \neg d\rangle & \langle\{d \Leftrightarrow b, \neg b\}, a \vee \neg d\rangle \\ \uparrow & \uparrow \\ \langle\{a \Rightarrow c, \neg c\}, \diamond\rangle & \langle\{a \vee b, a \Rightarrow c, \neg c\}, \diamond\rangle \end{array}\right.$$

$$\mathcal{C}\left\{\begin{array}{c} \langle\{a \vee b, a \Rightarrow c, \neg c, d \Leftrightarrow b\}, \neg(a \vee \neg d)\rangle \\ \uparrow \\ \langle\{\neg b\}, \diamond\rangle \end{array}\right.$$

FIG. 1.6 – Structure argumentative pour $a \vee \neg d$

$\diamondsuit$

Définition 1.31 (Catégoriseur, [BH01])**.** *Un* catégoriseur *associe des nombres aux arbres argumentatifs. Une catégorisation est alors une paire d'ensembles obtenue en appliquant le même catégoriseur à chacun des arbres argumentatifs d'une structure argumentative.*

Un catégoriseur tente de capturer les forces relatives des arguments en jeu pour ou contre une thèse initiale.

Les auteurs proposent un exemple de catégoriseur parmi d'autres : le *h-catégoriseur*, noté h. Un arbre de racine N est associé à :

$$h(N) = \frac{1}{1 + h(N_1) + \cdots + h(N_l)}$$

où $N_1, \ldots, N_l$ sont les fils du nœud N (si $l = 0$, $h(N_1) + \cdots + h(N_l) = 0$). Intuitivement, plus un argument possède de contrariants, plus sa valeur est petite. Récursivement, plus les contrariants de ses contrariants sont nombreux, moins la perte de valeur est importante.

D'autres catégoriseurs sont imaginés, comme le *binary-catégoriseur*, noté b, qui associe à chaque arbre la valeur 1 si toutes ses feuilles défendent la racine (voir exemple 1.13 sur le rétablissement), ou 0 sinon, ou encore le *c-catégoriseur* qui se contente de compter les arbres en leur attribuant à chacun la valeur 1.

Exemple 1.32 (suite de l'exemple 1.30)**.** Appliquons les différents catégoriseurs sur la structure argumentative obtenue dans l'exemple 1.30 :

- l'application du h-catégoriseur a pour résultat : $\big(\{1/2, 1/2\}, \{1/2\}\big)$;
- avec le binary-catégoriseur nous obtenons : $\big(\{0, 0\}, \{0\}\big)$;
- le c-catégoriseur aboutit à : $\big(\{1, 1\}, \{1\}\big)$.

$\diamondsuit$

Les valeurs produites par un catégoriseur sont ensuite « compilées » à l'aide d'un *accumulateur* afin de mesurer le crédit que l'on peut accorder à la formule se trouvant à l'origine de la structure argumentative considérée.

Définition 1.33 (Accumulateur, [BH01])**.** *Un* accumulateur *est une fonction qui prend en entrée une catégorisation pour une formule f et retourne une paire (f^+, f^-) où f^+ est la valeur accumulée pour f et f^- la valeur accumulée contre f. La balance des valeurs accumulées est $f^+ - f^-$.*

Si la balance des valeurs accumulées vaut 0, alors les arguments en faveur de la formule sont contrebalancés par ceux en sa défaveur. Si la balance est positive, les arguments en faveur de la formule sont prépondérants, et inversement.

Un accumulateur est proposé : le *c-accumulateur*, noté c_a, qui à partir d'une c-catégorisation comptabilise les arbres pour et contre une formule initiale.

Exemple 1.34 (suite et fin de l'exemple 1.32)**.** Appliquons le c-accumulateur sur la c-catégorisation obtenue : $c_a\big((\{1, 1\}, \{1\})\big) = (2, 1)$. La balance des valeurs accumulées vaut alors 1 $(= 2 - 1)$. Ainsi la thèse initiale qui est ici la formule a est retenue — la structure argumentative considérée est en faveur de a. $\diamondsuit$

Chapitre 2

Un outil : les X-logiques

La logique classique a depuis longtemps été considérée comme un outil privilégié en Intelligence artificielle pour la formalisation et l'exploitation des connaissances. Toutefois elle est inadéquate pour la représentation et le raisonnement avec des connaissances de sens commun, champ d'application de l'argumentation. Ces limites ont donné lieu à de nombreuses extensions appelées « logiques non classiques ». Parmi ces extensions on trouve des logiques dites *non-monotones* ([Bob80]) qui ne satisfont donc pas la propriété de monotonie — elle exprime le fait qu'un résultat déduit n'est jamais remis en cause.

Les X-logiques, introduites par Siegel et Forget dans [SF96], sont l'une d'elles. Elles n'avaient pas été utilisées auparavant au sein de systèmes argumentatifs, mais nous avons montré dans [Aub02], [AR05] et [AR06] qu'elles constituent une approche prometteuse pour la génération d'arguments. Le fait d'être pourvues d'un ensemble de formules paramétrant le procédé d'inférence classique a des répercussions essentielles tant sur la définition d'un agent que sur son comportement vis-à-vis d'un argument.

2.1 Définitions et propriétés fondamentales

Au départ les X-logiques se présentent comme une tentative de constitution d'une théorie la preuve pour les logiques non-monotones, à partir de la logique classique et d'un ensemble X de formules.

Remarque 2.1 (Inférence classique)**.** En logique classique, une formule f est un théorème de K si et seulement si son ajout à K ne change pas l'ensemble des conclusions dérivées de K, autrement dit :

$$K \vdash f \quad \text{ssi} \quad \overline{K \cup \{f\}} = \overline{K}$$

où la barre au dessus d'un ensemble dénote les théorèmes associés à l'ensemble en question[1]. $\diamondsuit$

Les X-logiques sont une généralisation — donc un affaiblissement — de la relation de conséquence classique $\vdash$:

Définition 2.2 (X-inférence)**.** *Soit X un ensemble de formules de $\mathcal{L}$ (X n'est pas nécessairement clos). La relation d'inférence $\vdash_X$ est définie par :*

$$K \vdash_X f \quad \textit{ssi} \quad \overline{K \cup \{f\}} \cap X \ \subseteq \ \overline{K}$$

Ainsi $K \vdash_X f$ si et seulement si tout théorème classique de $K \cup \{f\}$ qui est dans X est un théorème de K. Notons que l'ajout de f à K ne fait pas croître l'ensemble des théorèmes classiques qui sont dans X.

La propriété suivante correspond à la définition originale des X-logiques, telle qu'introduite dans [SF96]. Elle est équivalente à la définition 2.2.

Propriété 2.3 ([SF96])**.**

$$K \vdash_X f \quad \textit{ssi} \quad \overline{K \cup \{f\}} \cap X \ = \ \overline{K} \cap X$$

DÉMONSTRATION. De par la définition 2.2, $K \vdash_X f$ si et seulement si $\overline{K \cup \{f\}} \cap X \ \subseteq \ \overline{K}$. Or, si $\overline{K \cup \{f\}} \cap X$ est inclus dans $\overline{K}$, alors $\overline{K \cup \{f\}} \cap X \ \subseteq \ \overline{K} \cap X$. Par monotonie de l'inférence classique, $\overline{K} \cap X$ est inclus dans $\overline{K \cup \{f\}} \cap X$. Ainsi : $\overline{K \cup \{f\}} \cap X \ \subseteq \ \overline{K} \quad \text{ssi} \quad \overline{K \cup \{f\}} \cap X \ = \ \overline{K} \cap X$, et donc : $K \vdash_X f \quad \text{ssi} \quad \overline{K \cup \{f\}} \cap X \ = \ \overline{K} \cap X$. $\square$

La X-inférence n'est pas une inférence monotone. Si $A \vdash_X C$, il est possible d'avoir $A \cup B \nvdash_X C$. Par exemple : $\{a\} \vdash_{\{c\}} b \Rightarrow c$, mais : $\{a, b\} \nvdash_{\{c\}} b \Rightarrow c$.

Néanmoins, la comparaison entre la propriété 2.3 et la remarque 2.1 permet de bien se rendre compte que $\vdash_X$ est monotone sur X : si B est inclus dans X, alors $A \vdash_X B$ si et seulement si $A \vdash B$.

[1] Bien que cette notation puisse prêter à confusion en raison de sa similitude potentielle avec la notation usuelle du complémentaire d'un ensemble, elle correspond à la notation originale utilisée par [SF96] et permet d'alléger considérablement les écritures. Si besoin était, nous proposons de noter ici le complémentaire d'un ensemble E par E^c.

Propriété 2.4 (X-inférence, [Sad00])**.**

$$K \vdash_X f \quad ssi \quad \forall x \in X \setminus \overline{K},\ K \cup \{f\} \nvdash x$$

Intuitivement, l'ensemble de formules X permet de contraindre notre mode de raisonnement : un ensemble d'informations K entraîne f si l'addition de f à K ne produit pas plus de formules de X qu'avec K seulement. L'ensemble X apparaît comme un ensemble de formules *interdites*, dès lors que l'ensemble K ne les infère pas classiquement.

Lorsque X est l'ensemble de toutes les formules du langage $\mathcal{L}$, l'inférence $\vdash_X$ est la relation de conséquence classique $\vdash$. Si $X = \{\bot\}$, alors $K \vdash_X f$ est équivalent à $K \nvdash \neg f$ qui décrit la relation de consistance entre K et f — $K \wedge f$ est satisfiable — pourvu que K soit consistant par ailleurs. Si $X = \varnothing$, toutes les formules peuvent être déduites. Plus généralement, toutes formule peut être déduite lorsque X est inclus dans $\overline{K}$. En effet :

Propriété 2.5.

$$K \vdash_\varnothing f \quad ssi \quad \forall X \subseteq \overline{K},\ K \vdash_X f$$

Démonstration. Il suffit de constater que lorsque X est inclus dans $\overline{K}$, $X \setminus \overline{K}$ est l'ensemble vide, et par conséquent les deux inférences sont équivalentes. □

Nous utiliserons la terminologie suivante :

Définition 2.6 (Recevabilité)**.** *Soient K et X deux ensembles de formules.*

Une formule f est dite recevable par K au regard de X *si et seulement si $K \vdash_X f$. Elle est dite* non-recevable *dans le cas contraire ($K \nvdash_X f$).*

Par extension, un ensemble de formules E est recevable ou non-recevable par K au regard de X si et seulement si la conjonction des formules de E est respectivement recevable ou non-recevable par K au regard de X.

On a immédiatement les propriétés suivantes :

Propriétés 2.7.

1. (méta-cohérence) *Une formule ne peut être simultanément recevable et non-recevable.*
2. (paraconsistance) *Une formule et sa négation peuvent être simultanéments recevables.*
3. *Une formule et sa négation peuvent être simultanéments non-recevables.*

Exemples 2.8. Pour tout a et tout b :

- $\{a\} \vdash_{\{\bot\}} b$
- $\{b\} \vdash_{\{\bot\}} a \wedge b$, et : $\{b\} \vdash_{\{\bot\}} \neg(a \wedge b)$
- $\{b\} \nvdash_{\{\bot,a,\neg a\}} a \wedge b$, et : $\{b\} \nvdash_{\{\bot,a,\neg a\}} \neg(a \wedge b)$

$\diamondsuit$

2.2 Autres propriétés

Nous présentons ici d'autres propriétés qui caractérisent le comportement de ce formalisme. Dans [SF96], il est montré que les X-logiques vérifient les propriétés suivantes :

- Supra-classicité : $\dfrac{A \vdash B}{A \vdash_X B}$
- Réflexivité (*Ref*) : $A \vdash_X A$
- Cohérence ($\overline{X} \neq \{\top\}$) : $\top \nvdash_X \bot$
- Ou (*Or*) : $\dfrac{A \vdash_X C \quad B \vdash_X C}{A \vee B \vdash_X C}$
- Équivalence logique à gauche (*LLE*) : $\dfrac{\vdash A \Leftrightarrow B \quad A \vdash_X C}{B \vdash_X C}$
- Affaiblissement à droite (*RW*) : $\dfrac{\vdash B \Rightarrow C \quad A \vdash_X B}{A \vdash_X C}$
- Coupure (*Cut*) : $\dfrac{A \vdash_X B \quad A \wedge B \vdash_X C}{A \vdash_X C}$
- Cumulativité restreinte : $\dfrac{A \vdash B \quad A \vdash_X C}{A \wedge B \vdash_X C}$

Ces propriétés montrent que les X-logiques coïncident avec la notion d'*inférence permissive*, telle que définie par Bochman [Boc03].

D'autres propriétés toutes aussi importantes ne sont pas vérifiées dans le cas général par les X-logiques. Ce sont les propriétés *Et* et *Monotonie prudente* (*cf.* propriété 2.9). Leur point commun est qu'elles sont directement liées à la propriété de *Cumulativité*.

Propriété 2.9. *Pour tout ensemble de formules X, si l'ensemble des formules de $\mathcal{L}$ qui ne sont pas dans X est déductivement clos, alors la relation $\vdash_X$ a les propriétés suivantes :*

– *Et* (And) : $\dfrac{A \vdash_X B \quad A \vdash_X C}{A \vdash_X B \wedge C}$

– *Monotonie prudente* (CM) : $\dfrac{A \vdash_X B \quad A \vdash_X C}{A \wedge B \vdash_X C}$

– *Cumulativité* (Cut + CM) :

$$A \vdash_X B \quad \Rightarrow \quad (A \vdash_X C \textit{ ssi } A \wedge B \vdash_X C)$$

Exemple 2.10. Si le complémentaire de X n'est pas déductivement clos, on peut alors poser $X = \{a \wedge b \wedge c\}$, et avoir :
– $\{a\} \vdash_X b$
– $\{a\} \vdash_X c$

Mais :
– $\{a\} \nvdash_X b \wedge c$, donc la propriété *Et* n'est pas vérifiée,
– $\{a \wedge b\} \nvdash_X c$, et donc la propriété *Monotonie prudente* n'est pas non plus vérifiée.

$\diamondsuit$

2.3 Liens avec d'autres approches

Siegel et Forget montrent dans [SF96] que relations préférentielles et X-inférences sont liées :

Définition 2.11 (Relation préférentielle). *Une* relation préférentielle $\preccurlyeq$ *est une relation entre interprétations qui est réflexive ($i \preccurlyeq i$) et transitive (si $i \preccurlyeq j$ et $j \preccurlyeq k$, alors $i \preccurlyeq k$).*

Définition 2.12 (Modèle minimal). *Si A est un ensemble de formules, un* modèle minimal *de A est une interprétation i qui satisfait A ($i \vDash A$) et qui est minimale au regard de $\preccurlyeq$ et de l'ensemble des modèles de A (si i est un modèle minimal de A et si i' est un modèle de A tel que $i' \preccurlyeq i$, alors $i \preccurlyeq i'$).*

Définition 2.13 (Modèle préférentiel). *Au regard de la relation de préférence $\preccurlyeq$, un* modèle préférentiel *est défini par la relation $\vDash_\preccurlyeq$, telle que $A \vDash_\preccurlyeq B$ si et seulement si chaque modèle minimal de A satisfait B.*

Propriété 2.14 ([SF96]). *Quel que soit $\preccurlyeq$ une relation préférentielle, il existe un ensemble de formules X tel que $A \vDash_\preccurlyeq B$ ssi $A \vdash_X B$.*

Par conséquent, toute logique préférentielle est une X-logique. Une caractérisation de X est proposée dans [SF96] pour la circonscription avec prédicats fixes — résultat associé avec un résultat similaire donné dans un contexte différent par [Suc93]. Ce résultat est étendu au cas général de la circonscription avec prédicats libres dans [FS98]. Il est aussi montré que le raisonnement crédule (respectivement sceptique) au sein d'une théorie de défauts au sens de Reiter peut-être simulé en prenant pour X le complémentaire de l'union (respectivement de l'intersection) des extensions de cette théorie.

Chapitre 3

Des agents dotés d'attitudes

Plusieurs théories argumentatives considèrent la notion de *proposant-opposant* [Res77, Vre92] tandis que d'autres lui préfèrent la notion de système argumentatif au sein duquel des arguments construits à partir d'un même ensemble de formules s'enchaînent [LS89, Dun95, AC98, BH01]. Comme certains travaux [SL92], nous utiliserons une notion d'*agent*, proche du concept de proposant-opposant, mais permettant d'« attacher » des informations à chaque parti et s'inscrivant mieux dans une démarche dialectique :

Définition 3.1. *Un* agent *est un couple* $[K, X]$ *où* K *est une base de connaissances, et* X *un ensemble de formules contenant au moins* $\bot$*. L'*ensemble des agents*, sous-ensemble de* $2^{\mathcal{L}} \times 2^{\mathcal{L}}$*, est noté* $\mathcal{A}$*.*

Tandis que K est utilisé comme une représentation des connaissances d'un agent, X apparaît comme un ensemble de formules interdites pour le raisonnement de cet agent. La consistance des connaissances d'un agent et la présence systématique de la contradiction au sein de son ensemble de formules interdites impose à cet agent de n'accepter comme recevables que des formules consistantes avec ses connaissances. Ici la notion de recevabilité couvre donc au moins celle de consistance.

La répartition des formules d'un agent entre connaissances et interdits ne suit pas de règles préétablies. Si le choix peut parfois s'avérer délicat, certaines précisions de la définition peuvent trancher : les connaissances d'un agent doivent être consistantes. Sur ce point, l'ensemble des interdits offre une plus grande marge de manœuvre que celui des connaissances, et nous verrons qu'y placer une formule et sa négation est un bon moyen d'insuffler à l'agent certains « comportements » (exemples 3.19 sur la perplexité). Nous aurons l'occasion de revenir à plusieurs reprises sur cette question de répartition.

Ayant à l'esprit la définition à venir de la construction d'arguments par un agent, nous sommes particulièrement intéressés par les différents cas de figure envisageables face à la recevabilité d'une formule ou de sa négation. Ce sont ces situations que nous nommons *attitudes* d'un agent face à une formule. Ces attitudes dépendent donc simultanément des connaissances et des interdits de l'agent et amènent ainsi à étendre la recevabilité (définition 2.6) aux agents :

Définition 3.2. *Une formule f est dite* recevable *(respectivement* non-recevable*) par un agent $[K, X]$ si et seulement si f est recevable (respectivement non-recevable) par K au regard de X.*

Il découle de cette définition une autre distinction à faire entre connaissances et interdits d'un agent. Une formule f est recevable par un agent $[K, X]$ si $K \vdash_X f$. Or la propriété 2.4 sur les X-inférences montre clairement que si l'on s'intéresse à la clôture déductive de K lors du calcul de la recevabilité, il n'en est rien concernant X. Par conséquent, si l'on souhaite que certaines formules ne puissent être recevables par l'agent, et ceci uniquement par le biais d'interdits, alors il faudra toutes les expliciter dans X. Ce n'est toutefois pas la seule méthode.

Notons que nous ne faisons pas l'hypothèse du monde clos ([Rei78]), qui revient à dire que chaque littéral qui n'est pas affirmé est faux. Cette hypothèse nous apparaît comme trop contraignante car interdisant toute incertitude pour un agent à propos d'un fait (voir à ce propos la remarque 3.7).

3.1 Construction des attitudes

Au sens général, il est possible d'adopter plusieurs attitudes face à un fait. Ce dernier peut concorder avec nos connaissances et emporter notre accord, ou à l'opposé s'il va à leur encontre, nous seront tentés de nous positionner contre celui-ci. Mais il peut également arriver qu'aucun de ces deux choix nous conviennent, préférant statuer par exemple sur « je ne suis pas pour ce fait » sans pour autant être contre. Confronté à une alternative il se peut même que l'on ne sache se prononcer, les deux options nous semblant acceptables. Et encore, nous ne considérons là que les cas où la sincérité est de mise.

Comme étape préalable à la génération d'une réponse — au sens large — à un argument, un agent se doit donc d'analyser ce dernier en le comparant à ses propres connaissances et interdits. Les différentes attitudes que nous allons dégager permettront par la suite à cet agent de partitionner l'ensemble des formules du support de l'argument à l'aide d'opérateurs dûments définis (chapitre 4) — parce que les notions requises pour notre définition d'un argument ne sont pas encore exposées, nous considérons pour l'instant un argument comme un couple support-conclusion au sens de la définition 1.6.

Cela a été souligné lors de l'introduction aux X-logiques, nous pouvons parfaitement avoir simultanément pour K, X et f bien choisis, $K \vdash_X f$ et $K \vdash_X \neg f$ (propriété 2.7), c'est-à-dire qu'à la fois f et $\neg f$ seraient recevables par un agent $[K, X]$. De même, existe le cas de figure où une formule et sa négation sont simultanément non-recevables par un agent. L'exemple 3.9 illustrera ces deux possibilités.

En fait, au regard du caractère recevable ou non-recevable d'une formule ou de sa négation, les quatre situations élémentaires suivantes peuvent être envisagées : $K \vdash_X f$, $K \nvdash_X f$, $K \vdash_X \neg f$ ou encore $K \nvdash_X \neg f$. Notons-les a, $\mathsf{non\text{-}a}$, b et $\mathsf{non\text{-}b}$.

Théorème 3.3. *Exactement quatre cas de figure distincts peuvent être envisagés au regard de la recevabilité d'une formule et de sa négation :* (ab), $(\mathsf{a}\ \mathsf{non\text{-}b})$, $(\mathsf{non\text{-}a}\ \mathsf{non\text{-}b})$ *et* $(\mathsf{non\text{-}a}\ \mathsf{b})$.

Démonstration. Le point 1 de la propriété 2.7 permet de considérer a et b comme des variables booléennes. Déterminer les différents cas de figure envisageables se ramène alors à la recherche de toutes les interprétations d'un langage restreint à a et b. À partir du point 2 et 3 de la propriété 2.7 le résultat est immédiat.

□

Un agent confronté à une formule se retrouvera donc dans l'une des ces quatre situations.

Propriété 3.4. *Les ensembles* $\{\mathsf{a}, \mathsf{non\text{-}a}\}$, $\{(\mathsf{a}\ \mathsf{non\text{-}b}), (\mathsf{ab}), (\mathsf{non\text{-}a}\ \mathsf{b}), (\mathsf{non\text{-}a}\ \mathsf{non\text{-}b})\}$ *et* $\{\mathsf{b}, \mathsf{non\text{-}b}\}$ *constituent tous trois un partitionnement de l'ensemble des formules du langage.*

DÉMONSTRATION. Par construction, les situations a et non-a sont exclusives, les situations b et non-b le sont aussi. Le théorème 3.3 prouve que les combinaisons (a non-b), (ab), (non-a b) et (non-a non-b)} sont deux à deux distinctes. □

Le tableau 3.1 synthétise les relations ensemblistes entre les huit situations possibles : chaque ligne constitue un partitionnement de l'ensemble des formules du langage — si l'on n'est pas (a non-b) alors on est (ab), (non-a b) ou (non-a non-b) — et chaque colonne représente les imbrications — si l'on est (non-a b), on peut dire que l'on est aussi non-a ou encore b. Y changer de ligne revient ainsi à changer de partitionnement, tandis qu'y changer de colonne revient à changer de partition.

a		non-a	
a non-b	ab	non-a b	non-a non-b
non-b	b		non-b

TAB. 3.1 – Synthèse des relations inter-situations

La ligne intermédiaire du tableau 3.1 détermine le partitionnement de granularité la plus fine. Chacun des éléments de l'ensemble {(a non-b), (ab), (non-a b), (non-a non-b)} définit une *attitude* associée à un agent :

Définition 3.5 (Attitudes d'un agent). *Soient un agent $[K, X]$ et une formule f :*

- $[K, X]$ *est* pour f *ssi* $K \vdash_X f$ *et* $K \nvdash_X \neg f$
- $[K, X]$ *est* neutre *au sujet de* f *ssi* $K \vdash_X f$ *et* $K \vdash_X \neg f$
- $[K, X]$ *est* perplexe *au sujet de* f *ssi* $K \nvdash_X f$ *et* $K \nvdash_X \neg f$
- $[K, X]$ *est* contre f *ssi* $K \nvdash_X f$ *et* $K \vdash_X \neg f$

Par extension, un agent est pour, neutre, perplexe ou contre un ensemble de formules *si et seulement si cet agent est respectivement pour, neutre, perplexe ou contre la* conjonction des formules *de cet ensemble.*

3.2 Du nommage des attitudes

Expliquons-nous sur la façon dont nous avons nommé nos attitudes. Elle provient du fait que l'inférence $K \vdash_X f$ nous apparaît moins forte, moins porteuse d'informations que la non-inférence $K \nvdash_X f$.

En effet, si d'un côté nous pouvons justifier le fait que f est recevable par K au regard de X ($K \vdash_X f$) parce que f appartient aux théorèmes de K — auquel cas il existe un K' minimal, inclus dans K, permettant de

« motiver », de prouver l'inférence de f — il se peut, d'un autre côté, que l'inférence de cette formule ne soit autorisée que par défaut, ou dit autrement, par sa non-interaction avec K et X.

Par exemple posons $K = \{a, a \Rightarrow b, c\}$ et $X = \{\perp\}$. On a : $K \vdash_X b$. La recevabilité de la formule b peut se justifier par le fait que l'ensemble $\{a, a \Rightarrow b\}$ infère classiquement b. Mais on a également : $K \vdash_X d$. Et là, la recevabilité de la formule d ne se justifie que par la *neutralité* de d avec les connaissances et interdits en jeu. La formule d est en quelque sorte recevable par défaut.

Attendu qu'à terme nous désirons générer des arguments et comme cette neutralité nous y semble difficilement exprimable, la non-recevabilité nous apparaît plus intéressante que la recevabilité. Si f est non-recevable par K au regard de X, alors il y a obligatoirement des formules de K ou de X qui entrent en conflit avec f (*cf.* propriété 2.4 sur la X-inférence) :

$$\exists x \in X \setminus \overline{K}, \quad K \cup \{f\} \vdash x$$

Lorsqu'une personne se dit pour un fait il est implicitement entendu qu'elle possède des arguments en ce sens. C'est pourquoi nous souhaitons pouvoir attendre d'un agent pour une certaine formule qu'il soit en mesure de justifier sa position. La recevabilité simple n'en fournissant donc pas la garantie, nous nous tournons vers une formulation à base de non-recevabilité.

Un agent sera pour une formule f si $\neg f$ n'est pas recevable par lui (en fait il faut également que f soit recevable par l'agent afin de distinguer l'attitude pour de perplexe). Nous nous assurons ainsi que cet agent est muni d'informations en faveur de f.

De même, percevant la neutralité comme une absence d'interaction — du latin *neuter* « ni l'un ni l'autre » —, nous qualifions de neutre l'attitude « paraconsistante », c'est-à-dire celle déclarant simultanément recevables une formule et sa négation. Nous verrons que lors de la génération d'arguments (*cf.* section 6.2) cela correspond effectivement à n'avoir ni d'argument pour, ni d'argument contre.

Enfin nous déclarons perplexe un agent capable d'argumenter à la fois sur une formule et sa négation — du latin *perplexus* « embrouillé » —, et réservons l'attitude contre f à un agent pour $\neg f$ — c'est là l'une des propriétés des attitudes, *cf.* propriété 3.8.

Ainsi les attitudes sont nommées non pas par rapport à l'utilisation qu'elles font des X-inférences, mais par rapport aux arguments qui peuvent en être dégagés.

3.3 Propriétés intrinsèques

La définition 3.5 permet de se faire une idée un peu plus précise du processus de génération d'argument. Reprenons notre idée maîtresse. Lors d'une discussion, deux *agents* vont s'échanger des arguments. Le premier agent en avance un, c'est ensuite au tour du second de générer un argument en retour. Pour ce faire, ce dernier va analyser le support du premier argument — ce qu'il fera prochainement plus aisément par le biais des opérateurs de confrontation, section 4 — et en déduire qu'il est pour, neutre, perplexe ou contre certaines de ses parties. Si l'agent désire par exemple produire un argument attaquant le premier, il piochera l'une des parties envers lesquelles il est contre, et utilisera celle-ci pour une conclusion d'argument. Une fois la conclusion du futur argument arrêtée, il faudra générer son support. Nous verrons comment faire ceci en abordant la notion de réponse, chapitre 5.

Propriété 3.6. *Un agent est obligatoirement dans l'une et une seule des quatre attitudes possibles face à une formule.*

DÉMONSTRATION. La propriété 3.4 sur le partitionnement du langage conjuguée à la définition des attitudes d'un agent (définition 3.5) indique que les attitudes sont un partitionnement de l'ensemble des situations possibles face à une formule. □

Associant le tableau synthèse des relations entre situations (tableau 3.1) et la définition des attitudes d'un agent (définition 3.5), nous obtenons la classification du tableau 3.2 présentant les différentes attitudes envisageables par un agent face à une formule.

<table>
<tr><td colspan="2">f est recevable
$K \vdash_X f$</td><td colspan="2">f n'est pas recevable
$K \nvdash_X f$</td></tr>
<tr><td>Pour f
$K \nvdash_X \neg f$
et $K \vdash_X f$</td><td>Neutre au sujet de f
$K \vdash_X f$
et $K \vdash_X \neg f$</td><td>Contre f
$K \vdash_X \neg f$
et $K \nvdash_X f$</td><td>Perplexe au sujet de f
$K \nvdash_X f$
et $K \nvdash_X \neg f$</td></tr>
<tr><td>$\neg f$ n'est ...
$K \nvdash_X \neg f$</td><td colspan="2">$\neg f$ est recevable
$K \vdash_X \neg f$</td><td>... pas recevable
$K \nvdash_X \neg f$</td></tr>
</table>

TAB. 3.2 – Classification des attitudes envisageables par un agent $[K, X]$ face à une formule f

Remarque 3.7 (Hypothèse du monde clos)**.** Si nous faisions l'hypothèse du monde clos ([Rei78]), le degré de liberté de nos agents serait considérablement amoindri. En effet la propriété de paraconsistance de la X-inférence (propriété 2.7) disparaîtrait car $K \vdash_X f$ et $K \vdash_X \neg f$ implique :

$\forall x \in X \backslash \overline{K}, \quad K \cup \{\neg x\} \nvdash \neg f$ et $K \cup \{\neg x\} \nvdash f$. La situation « f est recevable par [K, X] » impliquerait alors « $\neg f$ n'est pas recevable par [K, X] » et « $\neg f$ est recevable par [K, X] » impliquerait « f n'est pas recevable par [K, X] ». Il en découlerait la classification présentée par le tableau 3.3.

Pour f	**Contre** f
$K \vdash_X f$	$K \vdash_X \neg f$

Tab. 3.3 – Classification des attitudes envisageables par un agent $[K, X]$ face à une formule f sous l'hypothèse du monde clos

◊

La classification 3.2 peut être représentée par l'*octaèdre des attitudes* (figure 3.1). Les trois lignes ou partitions de ce tableau y sont devenues les trois plans horizontaux passant par les sommets de l'octaèdre, tandis que les arêtes non horizontales précisent les changements de partitionnement — ou encore les implications lorsque l'on s'éloigne du plan médian.

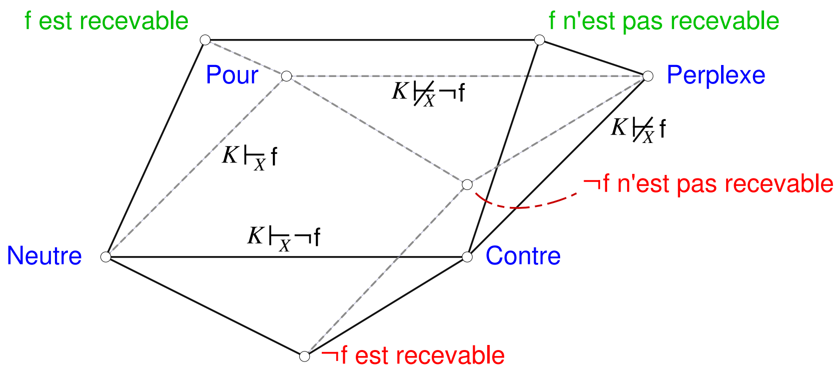


Fig. 3.1 – L'octaèdre des attitudes d'un agent $[K, X]$ face à une formule f

Il existe une interprétation évidente du plan médian de l'octaèdre dans la logique $\mathcal{FOUR}$ de Belnap ([Bel77]) : en considérant respectivement les quatre attitudes pour, neutre, contre et perplexe comme les valeurs *true*, $\top$, *false* et $\bot$ de cette logique, on reconstruit via la X-inférence le treillis complet (figure 3.2) proposé par Fitting dans [Fit91]. Toutefois, là où cette logique ne dispose que d'un opérateur de négation, nous distinguons évidemment négation d'une formule et négation de la recevabilité (non-recevabilité). De plus un agent n'adopte pas une attitude dans l'absolu, mais relativement à un ensemble de formules, ce qui rend plus complexe l'interprétation des attitudes comme de pures valeurs de vérité.

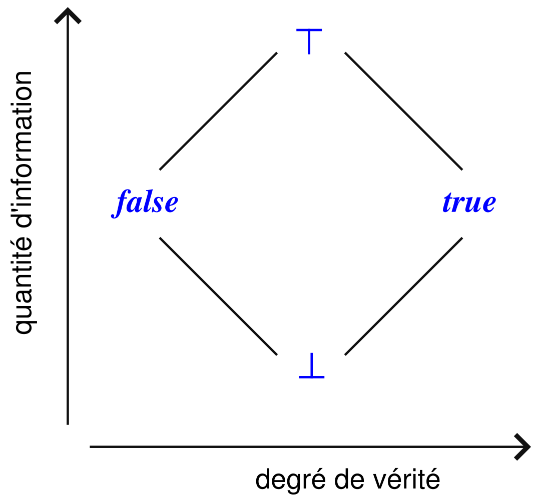


FIG. 3.2 – Treillis de la logique $\mathcal{FOUR}$

Propriété 3.8. *Soient un agent Φ et un ensemble de formules E :*
- *Φ est pour E ssi il est contre $\neg E$,*
- *Φ est neutre au sujet de E ssi il est neutre au sujet de $\neg E$,*
- *Φ est perplexe au sujet de E ssi il est perplexe au sujet de $\neg E$,*
- *Φ est pour les tautologies,*
- *Φ est contre les contradictions.*

DÉMONSTRATION. Elles se déduisent immédiatement de la définition des attitudes d'un agent (définition 3.5) par le biais du changement de variables. □

Exemple 3.9. Soit l'agent Φ possédant la base de connaissances suivante :

$$K_\Phi = \left\{ \begin{array}{l} \text{PierreVient} \Rightarrow \text{jeSuisGene}, \\ \neg\ \text{PierreVient} \Rightarrow \text{jeSuisTriste}, \\ \text{MarieVient} \Leftrightarrow \text{jeSuisContent} \end{array} \right\}$$

Nous allons tester l'attitude de cet agent vis-à-vis des formules suivantes, bien qu'il soit difficile de les imaginer au sein d'un même argument. Cela permettra de mettre en évidence le changement de comportement de notre agent relativement à son ensemble de formules interdites. Nous verrons par la suite comment un agent se prononce au sujet d'un argument qui lui est présenté (la section 6.3 en est un exemple illustré).

– PierreVient $\Rightarrow$ jeSuisGene ;
– PierreVient ;
– $\neg$ PierreVient ;
– MarieVient ;
– MarieVient $\wedge$ $\neg$ jeSuisContent.

Si l'agent possède comme seule contrainte d'être consistant avec ses connaissances — c'est-à-dire $X_\Phi = \{\perp\}$ — on a :

Pour :	{PierreVient $\Rightarrow$ jeSuisGene}
Neutre :	{PierreVient, $\neg$ PierreVient, MarieVient}
Contre :	{MarieVient $\wedge$ $\neg$ jeSuisContent}
Perplexe :	$\varnothing$

Maintenant nous ajoutons la formule « jeSuisGene » à l'ensemble d'interdits — $X_\Phi = \{\perp, \text{jeSuisGene}\}$ — nous obtenons *(en gras figurent les modifications)* :

Pour :	{PierreVient $\Rightarrow$ jeSuisGene, $\neg$ **PierreVient**}
Neutre :	{~~**PierreVient**, $\neg$ **PierreVient**,~~ MarieVient}
Contre :	{MarieVient $\wedge$ $\neg$ jeSuisContent, **PierreVient**}
Perplexe :	$\varnothing$

Interdisons enfin à l'agent « d'être triste ». Alors $X = \{\perp$, jeSuisGene, jeSuisTriste$\}$, et :

Pour :	{PierreVient $\Rightarrow$ jeSuisGene, ~~$\neg$ **PierreVient**~~}
Neutre :	{MarieVient}
Contre :	{MarieVient $\wedge$ $\neg$ jeSuisContent, ~~**PierreVient**~~}
Perplexe :	{**PierreVient**, $\neg$ **PierreVient**}

$\diamondsuit$

3.4 Évolution des attitudes d'un agent suivant ses interdits

La classification présentée par le tableau 3.2 est fonction de deux variables : d'une part l'agent considéré et d'autre part la formule sur laquelle il doit statuer. Nos agents ont l'originalité de posséder un ensemble de formules interdites. Nous pouvons donc nous interroger au sujet de l'influence de cet ensemble sur les attitudes que peut adopter un agent face à une formule donnée.

Intuitivement on peut s'attendre à ce que la réduction de l'ensemble de formules interdites d'un agent diminue le nombre de candidats — le nombre de formules — à l'attitude contre. Qu'en est-il réellement ? Et si l'on ajoute d'autres interdits ?

Théorème 3.10. *L'ensemble des formules du langage recevables par un agent* $[K, X]$ *croît si* X *décroît.*

DÉMONSTRATION. Soit X' strictement inclus dans X et soit f une formule :

- Si f est recevable par $[K, X]$, alors d'après la propriété d'une X-inférence (propriété 2.4) nous avons : $\forall x \in X \setminus \overline{K},\ K \cup \{f\} \nvdash x$. Donc : $\forall x \in X' \setminus \overline{K},\ K \cup \{f\} \nvdash x$. Ainsi f est aussi recevable par $[K, X']$.
- Si f n'est pas recevable par $[K, X]$, alors, toujours d'après la propriété d'une X-inférence, nous avons : $\exists x \in X \setminus \overline{K},\ K \cup \{f\} \vdash x$. Or x appartient soit à $X \cap X'$, soit à $X \setminus X'$. Dans le premier cas, f est non-recevable par l'agent $[K, X']$ et l'ensemble des formules du langage recevables par cet agent est inchangé, tandis que dans le second, f est recevable par $[K, X']$ et l'ensemble considéré croît.

Par conséquent, l'ensemble des formules recevables par $[K, X']$ contient — au sens large — l'ensemble des formules recevables par $[K, X]$. $\square$

Comme l'ensemble des formules du langage $\mathcal{L}$ recevables par un agent constitue avec l'ensemble des formules de $\mathcal{L}$ non recevables par ce même agent une partition de $\mathcal{L}$ (première ligne de la classification des attitudes, tableau 3.2), nous pouvons emplir le tableau 3.4 indiquant de façon schématique l'évolution de la taille des partitions de $\mathcal{L}$ relativement à un agent $[K, X]$ lorsque X décroît.

<table>
<tr><td colspan="2">f est recevable
$\nearrow$</td><td colspan="2">f n'est pas recevable
$\searrow$</td></tr>
<tr><td>Pour f</td><td>Neutre au sujet de f
$\nearrow$</td><td>Contre f</td><td>Perplexe au sujet de f
$\searrow$</td></tr>
<tr><td>$\neg f$ n'est ...
$\searrow$</td><td colspan="2">$\neg f$ est recevable
$\nearrow$</td><td>... pas recevable
$\searrow$</td></tr>
</table>

TAB. 3.4 – Évolution des partitions de la classification des attitudes envisageables par un agent $[K, X]$ face à une formule f lorsque X décroît

Il n'a été fait mention dans ce tableau d'aucune précision quant à l'évolution des partitions relatives aux attitudes pour et contre puisque ces dernières font toutes deux référence à la fois à la recevabilité d'une formule et à la non-recevabilité de la négation de cette formule. À ce stade nous pouvons quand même retenir que moins un agent possède d'interdits, plus il sera amené à se prononcer comme neutre au sujet de formules, tandis qu'il sera de moins en moins amené à se prononcer comme perplexe.

Approfondissons notre étude en examinant quelques cas limites, instantanés du tableau 3.4 suivant la valeur de X.

3.4.1 Cas limite : $X = \varnothing$

Propriété 3.11. *Tout agent $[K, \varnothing]$ est neutre au sujet de toute formule f.*

DÉMONSTRATION. Lorsque l'ensemble des formules interdites d'un agent est l'ensemble vide, toute formule est recevable par cet agent. Par conséquent les attitudes pour, contre et perplexe disparaissent puisqu'elles nécessitent l'existence de formules non recevables. Seule subsiste l'attitude « neutre au sujet de f », unifiée avec les situations « f est recevable » et « $\neg f$ est recevable ». □

Le tableau 3.5 reprend ce résultat.

f **est recevable**
=
$\neg f$ **est recevable**
=
Neutre au sujet de f
$K \vdash_\varnothing f$ et $K \vdash_\varnothing \neg f$

TAB. 3.5 – Classification des attitudes envisageables par un agent $[K, \varnothing]$ face à une formule f

3.4.2 Cas limite : card $(X) = 1$

Un agent dont l'ensemble des formules interdites est réduit à un singleton n'a accès qu'à trois des quatre attitudes.

Propriété 3.12. *Si un agent ne possède qu'un seul interdit alors il ne peut être perplexe au sujet d'une formule.*

DÉMONSTRATION. Supposons qu'un agent $[K, X]$, où X est un singleton, est perplexe au sujet d'une formule f. Alors la définition sur les attitudes d'un agent (définition 3.5) nous donne : $K \nvdash_X f$ et $K \nvdash_X \neg f$. Dit autrement : $\exists x_1 \in X \setminus \overline{K},\ K \cup \{f\} \vdash x_1$ et $\exists x_2 \in X \setminus \overline{K},\ K \cup \{\neg f\} \vdash x_2$. Comme X est un singleton, x_1 est égal à x_2, un unique x. Donc : $K \cup \{\neg x\} \vdash \neg f$ et $K \cup \{\neg x\} \vdash f$. On en déduit que $K \cup \{\neg x\}$ est inconsistant, et par conséquent que K infère x. Or x appartient à $X \setminus \overline{K}$, il y a donc contradiction. Ainsi un agent ne possèdant qu'une seule formule interdite ne peut être perplexe au sujet d'une formule. □

De cette propriété découle l'équivalence entre l'attitude « $[K, X]$ est pour f » et la situation « $\neg f$ est non-recevable par $[K, X]$ », ainsi qu'entre « $[K, X]$ est contre f » et « f est non-recevable par $[K, X]$ ». L'exemple 3.9 montrant qu'un tel agent peut toujours exprimer les attitudes pour, neutre et contre, nous aboutissons à la classification présentée par le tableau 3.6.

<table>
<tr><td colspan="2">f est recevable
$K \vdash_{\{x\}} f$</td><td rowspan="2">f n'est pas recevable
$=$
Contre f
$K \vdash_{\{x\}} \neg f$
ou $K \nvdash_{\{x\}} f$</td></tr>
<tr><td rowspan="2">Pour f
$=$
$\neg f$ n'est pas recevable
$K \nvdash_{\{x\}} \neg f$
ou $K \vdash_{\{x\}} f$</td><td>Neutre au sujet de f
$K \vdash_{\{x\}} f$
et $K \vdash_{\{x\}} \neg f$</td></tr>
<tr><td colspan="2">$\neg f$ est recevable
$K \vdash_{\{x\}} \neg f$</td></tr>
</table>

TAB. 3.6 – Classification des attitudes envisageables par un agent $[K, \{x\}]$ face à une formule f

Notons qu'en choisissant $X = \{\bot\}$ nous tombons sur la relation de consistance entre K et f — $K \vdash_X f$ est equivalent à $K \nvdash \neg f$, pourvu que K soit consistant. Un agent avec pour seule contrainte d'être cohérent ne peut donc se retrouver dans l'attitude perplexe.

3.4.3 Cas limite : $X = \mathcal{L}$

Lorsque $X = \mathcal{L}$, $\vdash_X$ est la relation de conséquence classique $\vdash$. Comme K est consistant — base de connaissances d'un agent —, nous en déduisons que $K \vdash_{\mathcal{L}} f$ implique $K \nvdash_{\mathcal{L}} \neg f$. Il en résulte la disparition de l'attitude neutre et l'équivalence entre l'attitude « $[K, X]$ est pour f » et la situation « f est recevable par $[K, X]$ », ainsi qu'entre « $[K, X]$ est contre f » et « $\neg f$ est recevable par $[K, X]$ ». La classification des attitudes (tableau 3.2) évolue alors pour devenir la classification 3.7.

Ce résultat est en adéquation avec le fait que la X-inférence généralise l'inférence classique : la classification des attitudes produite en utilisant une inférence classique — c'est-à-dire en choisissant $X = \mathcal{L}$ — est incluse dans celle produite en exploitant les X-logiques.

<table>
<tr><td colspan="2">f n'est pas recevable
$K \nvdash_{\mathcal{L}} f$</td><td rowspan="2">Pour f
$=$
f est recevable
$K \vdash_{\mathcal{L}} f$</td></tr>
<tr><td rowspan="2">Contre f
$=$
$\neg f$ est recevable
$K \vdash_{\mathcal{L}} \neg f$</td><td>Perplexe au sujet de f
$K \nvdash_{\mathcal{L}} f$
et $K \nvdash_{\mathcal{L}} \neg f$</td></tr>
<tr><td colspan="2">$\neg f$ n'est pas recevable
$K \nvdash_{\mathcal{L}} \neg f$</td></tr>
</table>

TAB. 3.7 – Classification des attitudes envisageables par un agent $[K, \mathcal{L}]$ face à une formule f

3.5 Évolution des attitudes d'un agent au sujet d'un ensemble croissant

Si un agent est pour, neutre, contre ou perplexe au sujet d'un ensemble de formules, qu'advient-il lorsque l'on y ajoute une formule supplémentaire ? Cette question est non seulement intéressante si l'on considère l'implémentation et donc la complexité associée à l'analyse d'un argument adverse, mais peut également être source de subtilités au sein de stratégies argumentatives.

Nous présentons tout d'abord deux lemmes nécessaires aux démonstrations des propriétés à suivre.

Lemme 3.13. *Soient un agent Φ et A et B deux ensembles quelconques de formules. Si A n'est pas recevable par Φ alors $A \cup B$ n'est pas recevable par Φ.*

DÉMONSTRATION. Posons $\Phi = [K, X]$. Si : $K \nvdash_X A$, alors : $\exists x \in X \setminus \overline{K},\ K \cup A \vdash x$. Donc : $\exists x \in X \setminus \overline{K},\ K \cup A \cup B \vdash x$, et ainsi : $K \nvdash_X A \cup B$. $\square$

Lemme 3.14. *Soient un agent Φ et A et B deux ensembles quelconques de formules. Si $\neg A$ est recevable par Φ alors $\neg(A \cup B)$ est recevable par Φ.*

DÉMONSTRATION. Posons $\Phi = [K, X]$. Par hypothèse : $K \vdash_X \neg A$, donc : $\forall x \in X \setminus \overline{K},\ K \cup \{\neg A\} \nvdash x$. Or : $K \cup \{\neg A\} \vdash (K \cup \{\neg A\}) \vee (K \cup \{\neg B\})$. Alors : $\forall x \in X \setminus \overline{K},\ K \cup \{\neg A \vee \neg B\} \nvdash x$, ce qui revient à dire que : $\forall x \in X \setminus \overline{K},\ K \cup \{\neg(A \cup B)\} \nvdash x$. Par conséquent : $K \vdash_X \neg(A \cup B)$. $\square$

Ces deux lemmes font état d'une notion de *stabilité* : dès lors qu'un ensemble est non-recevable par un agent, il gardera ce statut quelles que soient les formules qu'on lui ajoute, et pareillement au sujet de la négation d'un ensemble.

Les attitudes d'un agent, recourant dans leur définition (définition 3.5) tantôt à la non-recevabilité d'un ensemble, tantôt à la recevabilité de la négation d'un ensemble, montreront ainsi plus ou moins de stabilité face à la croissance de cet ensemble.

Propriété 3.15 (Contre). *Si un agent Φ est contre un ensemble A, alors cet agent est contre tout ensemble contenant A.*

DÉMONSTRATION. Si un agent Φ est contre A, alors A n'est pas recevable par Φ. Donc d'après le lemme 3.13 $A \cup B$ ne le sera pas non plus. Si un agent Φ est contre A, alors $\neg A$ est recevable par Φ. Donc d'après le lemme 3.14 $\neg(A \cup B)$ le sera aussi. Par conséquent Φ sera contre $A \cup B$. □

Propriété 3.16 (Neutre). *Si un agent Φ est neutre au sujet d'un ensemble A, alors cet agent est soit neutre, soit contre tout ensemble contenant A.*

DÉMONSTRATION. Si un agent Φ est neutre au sujet de A, alors $\neg A$ est recevable par Φ. Donc d'après le lemme 3.14, $\neg(A \cup B)$ le sera aussi. Cela restreint le nombre d'attitudes possibles à deux : neutre ou contre (*cf.* classification des attitudes, tableau 3.2). □

Les exemples suivants montrent que les deux cas sont possibles :

Exemples 3.17. Dans chacun de ces exemples, Φ est neutre au sujet de A.
- $\Phi = [\{b\}, \{\bot\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est neutre au sujet de $A \cup B$ (et pour B).
- $\Phi = [\varnothing, \{\bot, \neg b, a \wedge b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est contre $A \cup B$ (et pour B).
- $\Phi = [\varnothing, \{\bot\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est neutre au sujet de $A \cup B$ (et neutre au sujet de B).
- $\Phi = [\varnothing, \{\bot, a \wedge b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est contre $A \cup B$ (et neutre au sujet de B).

◇

Propriété 3.18 (Perplexe). *Si un agent Φ est perplexe au sujet d'un ensemble A, alors cet agent est soit perplexe, soit contre tout ensemble contenant A.*

Démonstration. Si Φ est perplexe au sujet de A, alors A n'est pas recevable par Φ. Donc d'après le lemme 3.13 $A \cup B$ ne le sera pas non plus. Cela restreint le nombre d'attitudes possibles à deux : perplexe ou contre (*cf.* classification des attitudes, tableau 3.2). □

Là aussi, les deux cas sont possibles. Illustration par l'exemple :

Exemples 3.19. Dans chacun de ces exemples, Φ est perplexe au sujet de A.

- $\Phi = [\{b\}, \{\perp, a, \neg a\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est perplexe au sujet de $A \cup B$ (et pour B).
- $\Phi = [\varnothing, \{\perp, a, \neg a, \neg b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est contre $A \cup B$ (et pour B).
- $\Phi = [\varnothing, \{\perp, a, \neg a, b, \neg b, \neg a \vee \neg b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est perplexe au sujet de $A \cup B$ (et perplexe au sujet de B).
- $\Phi = [\varnothing, \{\perp, a, \neg a, b, \neg b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est contre $A \cup B$ (et perplexe au sujet de B).

♢

Propriété 3.20 (Pour)**.** *Même si un agent Φ est pour les ensembles de formules A et B, on ne peut rien présager de son attitude vis-à-vis de l'ensemble $A \cup B$.*

Les exemples suivants montrent que tous les cas sont possibles :

Exemples 3.21. Dans chacun de ces exemples l'agent Φ est à la fois pour A et pour B :

- $\Phi = [\{a, b\}, \{\perp\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est pour $A \cup B$.
- $\Phi = [\varnothing, \{\perp, \neg a, \neg b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est neutre au sujet de $A \cup B$ (notamment car $\neg(a \wedge b)$ n'implique ni $\neg a$ ni $\neg b$).
- $\Phi = [\varnothing, \{\perp, \neg a, \neg b, a \wedge b\}]$, $A = \{a\}$ et $B = \{b\}$. Φ est contre $A \cup B$ (même remarque que ci-dessus).
- $\Phi = [\{a\}, \{\perp, b \wedge c, \neg b \vee \neg c\}]$, $A = \{a \Rightarrow b\}$ et $B = \{a \Rightarrow c\}$. Φ est perplexe au sujet de $A \cup B$ (on notera que $A \cup B$ est équivalent à $\neg a \vee (b \wedge c)$).

♢

Propriété 3.22 (Neutre et perplexe)**.** *Si un agent Φ est neutre au sujet d'un ensemble de formules A et perplexe au sujet d'un ensemble de formules B, alors cet agent est contre $A \cup B$.*

DÉMONSTRATION. Si un agent Φ est perplexe au sujet de B, alors B n'est pas recevable par Φ. Donc d'après le lemme 3.13 $A \cup B$ ne l'est pas non plus. Si un agent Φ est neutre au sujet de A, alors $\neg A$ est recevable par Φ. Donc d'après le lemme 3.14 $\neg(A \cup B)$ l'est aussi. Par conséquent Φ est contre $A \cup B$. $\square$

Corollaire 3.23. *Étant donnés deux ensembles A et B de formules, le tableau 3.8 énumère l'ensemble des attitudes qu'un agent peut adopter au sujet de $A \cup B$ en fonction de son attitude au sujet de A et de son attitude au sujet de B.*

DÉMONSTRATION. Dans l'ordre de lecture.

1. Pour A et pour B : d'après la propriété 3.20 toute attitude est envisageable pour $A \cup B$.
2. Pour A et neutre au sujet de B :
   - le premier et le deuxième des exemples 3.17 confirment que l'on peut être respectivement neutre ou contre $A \cup B$;
   - la propriété 3.16 prouve que l'on ne peut aboutir à d'autres attitudes.
3. Pour A et perplexe sur B :
   - le premier et le deuxième des exemples 3.19 garantissent que l'on peut être respectivement perplexe ou contre $A \cup B$;
   - la propriété 3.18 démontre que l'on ne peut évoluer vers une autre attitude.
4. Pour A et contre B : la propriété 3.15 affirme que l'on ne peut qu'être contre $A \cup B$.
5. Neutre au sujet de A et de B :
   - le troisième et le dernier des exemples 3.17 montrent que l'ont peut être respectivement neutre ou contre $A \cup B$;
   - la propriété 3.16 prouve que l'on ne peut aboutir à d'autres attitudes.
6. Neutre au sujet de A et perplexe sur B : la propriété 3.22 démontre que l'on ne peut qu'être contre $A \cup B$.
7. Neutre au sujet de A et contre B : *cf.* 4.
8. Perplexe sur A et sur B :
   - le troisième et le dernier des exemples 3.19 attestent que l'on peut être respectivement perplexe ou contre $A \cup B$;

   – la propriété 3.18 prouve que l'on ne peut évoluer vers une autre attitude.

9. Perplexe sur A et contre B : *cf.* 4.
10. Contre A et contre B : *cf.* 4.

$\square$

<table>
<tr><th colspan="3">Attitude d'un agent</th></tr>
<tr><th>vis-à-vis de A</th><th>vis-à-vis de B</th><th>vis-à-vis de $A \cup B$</th></tr>
<tr><td rowspan="4">Pour</td><td>Pour</td><td>toutes possibles</td></tr>
<tr><td>Neutre</td><td>Neutre ou Contre</td></tr>
<tr><td>Perplexe</td><td>Perplexe ou Contre</td></tr>
<tr><td>Contre</td><td>Contre</td></tr>
<tr><td rowspan="3">Neutre</td><td>Neutre</td><td>Neutre ou Contre</td></tr>
<tr><td>Perplexe</td><td rowspan="2">Contre</td></tr>
<tr><td>Contre</td></tr>
<tr><td rowspan="2">Perplexe</td><td>Perplexe</td><td>Perplexe ou Contre</td></tr>
<tr><td>Contre</td><td>Contre</td></tr>
<tr><td>Contre</td><td>Contre</td><td>Contre</td></tr>
</table>

TAB. 3.8 – Évolution des attitudes d'un agent au sujet d'un ensemble de formules croissant

La figure suivante (figure 3.3) est issue du tableau 3.8. Elle exhibe le fait que plus un ensemble croît, plus l'agent s'y confrontant tend vers l'attitude contre, attitude que l'agent ne pourra plus alors quitter. C'est la raison pour laquelle on la trouve en bas de cette figure qui ordonne de façon croissante les attitudes suivant une notion de *stabilité*. L'attitude pour y figure à l'opposé car souhaiter la maintenir tout en augmentant l'ensemble oblige à n'inclure dans ce dernier que des ensembles envers lesquels l'agent est également pour. Et encore cette condition n'est pas suffisante (*cf.* propriété 3.20).

De façon anectodique, dans un graphe dont l'axe des abscisses mesure la croissance d'un ensemble de formules et dont l'axe des ordonnées énumère les diverses attitudes du pour au contre, la courbe de l'évolution de l'attitude d'un agent face à cet ensemble est systématiquement croissante (non strictement).

Est-ce à dire qu'une stratégie argumentative prudente, cherchant à éviter les conflits, serait de ne construire que de petits arguments ?

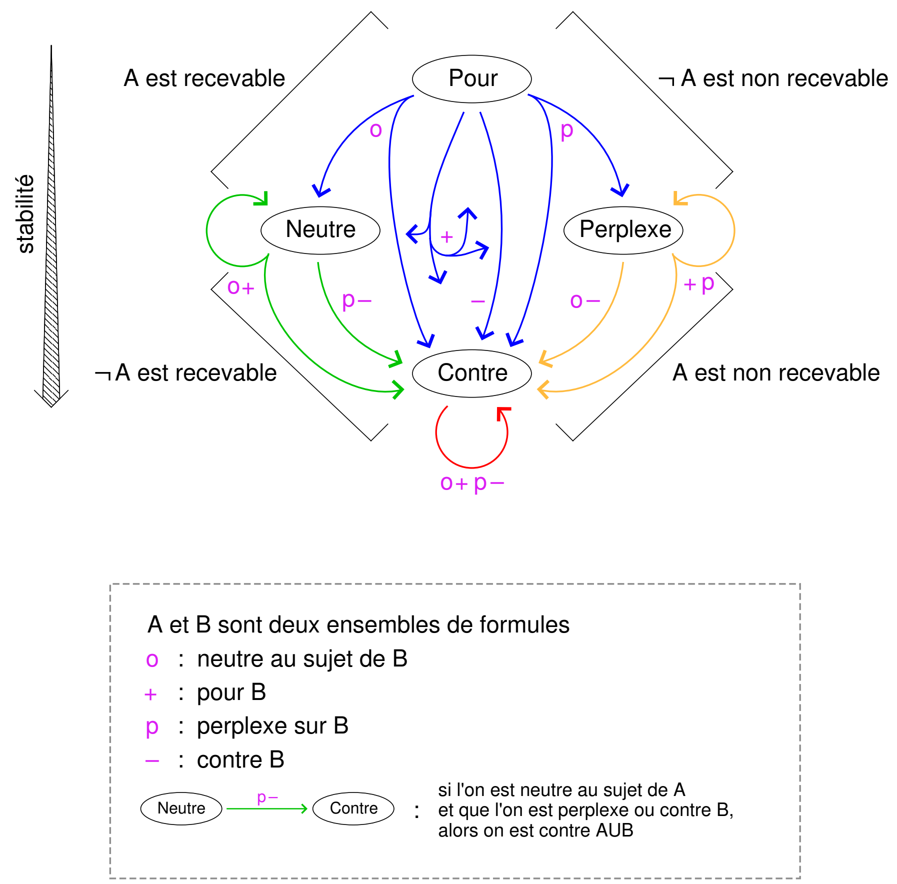


FIG. 3.3 – Évolution des attitudes d'un agent au sujet d'un ensemble de formules croissant

Chapitre 4

Opérateurs de confrontation

Lors d'une discussion, deux agents vont s'échanger des arguments. Précisons à nouveau que même si nous n'en avons pas encore présenté notre définition, nous partageons la conception d'un argument vu comme un couple support-conclusion, c'est-à-dire au sens de la définition 1.6. Le premier agent avance un argument. C'est ensuite au tour du second de générer un argument en retour. La conclusion de celui-ci est à choisir en fonction des attitudes du deuxième agent par rapport au premier argument : s'il n'est pas d'accord sur une partie de l'argument, il peut orienter sa conclusion en ce sens, ou au contraire souligner grâce à celle-ci un point d'entente. Les *opérateurs de confrontation* doivent permettre de déterminer les attitudes de l'agent vis-à-vis des différentes parties de l'argument extérieur.

4.1 Définition

Définition 4.1. *L'*opérateur $|_+$ *associe à un agent* $[K, X]$ *et un ensemble de formules* E *les* sous-ensembles de formules *de* E *tels que cet agent est pour chacun d'eux :*

$$\begin{array}{llcl} |_+ : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \,|_+\, E & \longmapsto & \{P \subseteq E \mid K \vdash_X P \text{ et } K \nvdash_X \neg P\} \end{array}$$

De la même façon nous définissons les opérateurs $|_0$, $|_-$ *et* $|_p$ *qui per-*

mettent d'accéder respectivement aux ensembles des formules au sujet desquels un agent est neutre, contre ou perplexe :

$$\begin{array}{llll} |_0 : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \mid_0 E & \longmapsto & \{P \subseteq E \mid K \vdash_X P \ \textit{et} \ K \vdash_X \neg P\} \end{array}$$

$$\begin{array}{llll} |_- : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \mid_- E & \longmapsto & \{P \subseteq E \mid K \nvdash_X P \ \textit{et} \ K \vdash_X \neg P\} \end{array}$$

$$\begin{array}{llll} |_p : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \mid_p E & \longmapsto & \{P \subseteq E \mid K \nvdash_X P \ \textit{et} \ K \nvdash_X \neg P\} \end{array}$$

Ces opérateurs fournissent un moyen aisé de choisir une conclusion pour un futur argument : qu'un agent désire par exemple produire un argument pour ou contre tout ou partie de l'argument qui lui est présenté, alors la conclusion qui lui convient est à prendre dans l'ensemble produit par les opérateurs $|_+$ ou $|_-$ appliqués sur le support de l'argument extérieur. L'illustration constituant la section 6.3 fournit un exemple de ce processus.

Propriété 4.2. *Quels que soient l'agent Φ et l'ensemble de formules E considérés, les ensembles produits par ces quatre opérateurs de confrontation sont deux à deux distincts :*

$$\forall *, \star \in \{+, 0, p, -\}, \quad \Phi \mid_* E \ \cap \ \Phi \mid_\star E \ = \ \varnothing$$

Démonstration. Conséquence immédiate de la propriété 3.6 traitant du partitionnement du langage vis-à-vis des attitudes. □

Propriété 4.3. *Quels que soient l'agent Φ et l'ensemble de formules E considérés, E est contenu dans l'union des ensembles produits par les quatre opérateurs de confrontation :*

$$E \subseteq \bigcup_{* \in \{+,0,p,-\}} \Phi \mid_* E$$

Démonstration. De la propriété 3.6 traitant du partitionnement du langage vis-à-vis des attitudes, on déduit que pour toute formule f de l'ensemble E, $\{f\}$ appartient à l'ensemble généré par l'un de ces quatre opérateurs. Par conséquent E est inclus dans leur union. □

4.2 Construction des opérateurs de confrontation

De la même façon que les attitudes sont construites à partir des quatre utilisations possibles d'une X-inférence ou de sa négation, les opérateurs de confrontation peuvent être élaborés à partir d'*opérateurs de base* n'utilisant qu'une seule inférence. Ainsi il y aurait au plus quatre de ces opérateurs de base, mais deux suffisent s'ils sont choisis judicieusement. En effet, comme on ne peut à la fois inférer et non inférer une formule, chacun forme avec l'un des trois autres une partition de 2^E. En prendre deux qui ne s'apparient pas de la sorte fournit alors deux briques élémentaires capables de construire les quatre opérateurs de confrontation.

Suite à la discussion (section 3.2) sur le nommage des attitudes et faisant part de notre préférence pour la non-recevabilité sur la recevabilité, nous proposons dès à présent un premier opérateur de base :

Définition 4.4. *L'*opérateur de non-recevabilité *noté* $|_{\not r}$ *associe à un agent* $[K, X]$ *et un ensemble de formules* E *les* sous-ensembles de formules *de* E *tels que chacun d'eux est non recevable par* $[K, X]$ *:*

$$\begin{array}{llll} |_{\not r} : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \,|_{\not r}\, E & \longmapsto & \{P \subseteq E \mid K \nvdash_X P\} \end{array}$$

Pour autant, la section traitant de la génération d'arguments (section 6.2) montrera que l'opérateur de non-recevabilité ne nous limite pas à la génération d'arguments contestataires. La première des propriétés 3.8 nous indique déjà pourquoi : Φ *est pour* A *si et seulement si il est contre* $\neg A$.

Propriété 4.5. *L'ensemble d'ensembles envers lesquels un agent* Φ *est perplexe ou contre est l'ensemble d'ensembles non-recevables par cet agent.*

$$\Phi \,|_{\not r}\, E \;=\; \Phi \,|_{-}\, E \,\cup\, \Phi \,|_{p}\, E$$

Démonstration. Par construction et conformément à la classification des attitudes (tableau 3.2). □

Il suffit d'un seul autre opérateur de base pour être capable de générer les quatre opérateurs de confrontation. Deux choix sont possibles :

Définition 4.6. *Les opérateurs notés* $|_y$ *et* $|_z$ *associent à un agent* $[K, X]$ *et un ensemble de formules* E *les* sous-ensembles de formules P *de* E *tels que pour chacun d'eux* $\neg P$ *est respectivement recevable et non-recevable par*

$[K, X]$.

$$\begin{array}{lllll} |_y : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \,|_y\, E & \longmapsto & \{P \subseteq E \mid K \vdash_X \neg P\} \end{array}$$

$$\begin{array}{lllll} |_z : & \mathcal{A} \times 2^{\mathcal{L}} & \longrightarrow & 2^{2^{\mathcal{L}}} \\ & [K, X] \,|_z\, E & \longmapsto & \{P \subseteq E \mid K \nvdash_X \neg P\} \end{array}$$

Choisissons arbitrairement l'opérateur $|_y$. Il est alors possible de construire les quatre opérateurs de confrontation à partir des opérateurs de base $|_{\not y}$ et $|_y$:

Propriété 4.7 (Construction)**.** *Soient* Φ *un agent et* E *un ensemble de formules :*

$$\begin{array}{lcl} \Phi \,|_+\, E & = & 2^E \setminus \left(\Phi \,|_{\not y}\, E \,\cup\, \Phi \,|_y\, E\right) \\ \Phi \,|_0\, E & = & \Phi \,|_y\, E \,\setminus\, \Phi \,|_{\not y}\, E \\ \Phi \,|_-\, E & = & \Phi \,|_{\not y}\, E \,\cap\, \Phi \,|_y\, E \\ \Phi \,|_p\, E & = & \Phi \,|_{\not y}\, E \,\setminus\, \Phi \,|_y\, E \end{array}$$

DÉMONSTRATION. Par construction et conformément à la classification des attitudes (tableau 3.2). □

La figure 4.1 illustre ces propriétés. Il est à noter que l'on serait arrivé à un résultat similaire en ayant préféré l'opérateur de base $|_z$ à $|_y$.

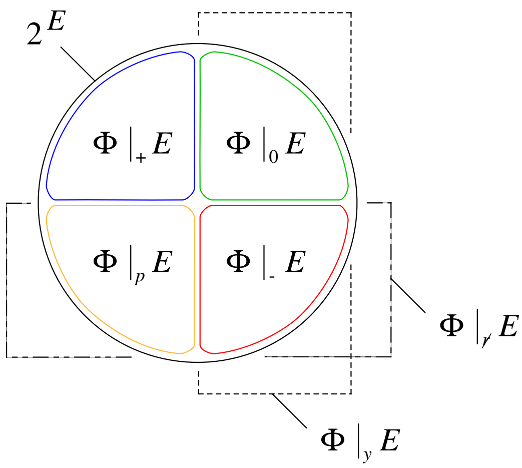


FIG. 4.1 – Résultat de la confrontation entre un agent Φ et un ensemble de formules E

Chapitre 5

Des réponses comme supports d'arguments

Une question primordiale lors de la génération d'un argument est : « Dans quel but ? ». L'objectif est de présenter cet argument au sein d'une discussion entre agents. Il doit donc, schématiquement, apporter sa contribution en défendant ou désapprouvant certains des faits avancés.

On s'intéresse au processus de fabrication d'un argument à partir à la fois d'un ensemble de formules et des attitudes d'un agent face à cet ensemble. L'idée est de lier agent et argument via la notion de *réponse* définie ci-après. Intuitivement une réponse est l'ensemble de formules qui a « motivé » l'attitude de l'agent.

5.1 Réponses et propriétés

Une réponse peut et doit dans certains cas puiser dans l'ensemble des formules interdites de cet agent, éventuellement en plus d'un recours à sa base de connaissances, afin d'être en mesure de justifier l'attitude de ce dernier. Pour autant, ces attitudes sont définies sur la recevabilité ou la non-recevabilité de formules (*cf.* classification des attitudes, tableau 3.2). Il apparaît donc comme nécessaire de corréler l'exploitation des interdits d'un agent au sein d'une réponse à leur utilisation faite par la X-inférence, à la base de la notion de recevabilité. C'est pourquoi la définition d'une réponse doit intégrer la notion de X-inférence.

Par ailleurs, lors de la construction des attitudes, nous avons fait état d'une *dissymétrie* entre recevabilité et non-recevabilité (*cf.* section 3.2 sur le nommage des attitudes) : si une formule f est non-recevable par l'agent $[K, X]$, alors il existe obligatoirement des formules de K ou de X qui entrent en conflit avec f, tandis qu'aucune interaction n'est garantie lors de la recevabilité d'une formule. Attendu que nous souhaitons faire des réponses la « base » des arguments, leur matière première, et qu'y placer l'information « rien ne contredit f, donc on a f » nous ferait sortir du cadre de construction monotone des arguments — à l'instar des systèmes argumentatifs présentés dans ce manuscrit, un argument reste toujours valide même si de nouvelles connaissances sont ajoutées, la non-monotonie apparaît seulement en termes d'interactions entre arguments conflictuels — nous choisissons de définir les réponses sur l'idée de non-recevabilité.

Si un ensemble est non-recevable par un agent, nous sommes assurés que ce dernier a matière à argumenter. Lorsque l'on écrit : $K \nvdash_X A$, cela signifie (propriété 2.4) : $\exists x \in X \setminus \overline{K},\ K \cup A \vdash x$. Mais cela peut se réécrire de la sorte : $\exists x \in X \setminus \overline{K},\ K \cup \{\neg x\} \vdash \neg A$. C'est en ce sens que l'on exploite base de connaissance et interdits d'un agent pour construire une réponse — ici $K \cup \{\neg x\}$ — à un ensemble de formules — ici A. Cette réponse deviendra la justification d'un argument dont la conclusion pourrait être $\neg A$.

Définition 5.1 (Réponse). *Une* réponse *d'un agent* $[K, X]$ *à un ensemble consistant de formules* A *est un ensemble consistant de formules* R *tel que, étant donnés* $K' \subseteq K$ *et* $X' \subseteq X$ *:*

1. $R = K' \cup (\bar{\cup} X')$
2. $K' \nvdash_{\{\bot\} \cup X'} A$

L'ensemble des réponses de $[K, X]$ *à* A *est noté* $\mathcal{R}^A_{[K,X]}$.

Beaucoup de choses sont à dire sur cette définition. Commençons par quelques remarques :

a. Il se peut fort bien que l'intersection $K' \cap (\bar{\cup} X')$ soit non vide, mais ceci est sans incidence. La propriété 5.2 traitant de la réduction de l'ensemble d'interdits en montrera la raison.

b. Nous ne contraignons pas X' à appartenir à $X \setminus \overline{K}$. Nous verrons plus loin que ce n'est pas anodin (voir section 5.4 traitant du mensonge, et plus particulièrement le corollaire 5.17).

c. La condition 2 peut laisser penser que la notion de réponse n'est vue qu'au travers d'une démarche éristique. Rappelons que notre propriété 3.8 affirme qu'être contre A est équivalent à être pour $\neg A$.

d. Enlever la contradiction de la condition 2 de la définition — c'est-à-dire avoir $K' \nvdash_{X'} A$ au lieu de $K' \nvdash_{\{\bot\}\cup X'} A$ — impliquerait le fait qu'une réponse ne pourrait plus se construire uniquement sur les connaissances de l'agent. Par exemple, $\{a, a \Rightarrow b\}$ ne serait plus une réponse de l'agent $[\{a, a \Rightarrow b\}, \{\bot\}]$ à $\{\neg b\}$, contrairement à $\{a, a \Rightarrow b, \top\}$. L'introduction de la contradiction permet de générer des réponses plus naturelles.

Cette définition pose le problème de la décomposition de R sur K et X : il se peut fort bien que l'intersection de K avec $\overset{\neg}{\cup}X$ soit non nulle, permettant le cas échéant plusieurs possibilités de constructions d'une même réponse R. L'intersection entre K, $\overset{\neg}{\cup}X$ et R est notée I sur la figure 5.1, et d'après la définition d'une réponse, chaque possibilité de répartition des formules de I entre K et $\overset{\neg}{\cup}X$ donne lieu à une décomposition valide de R sur K et X.

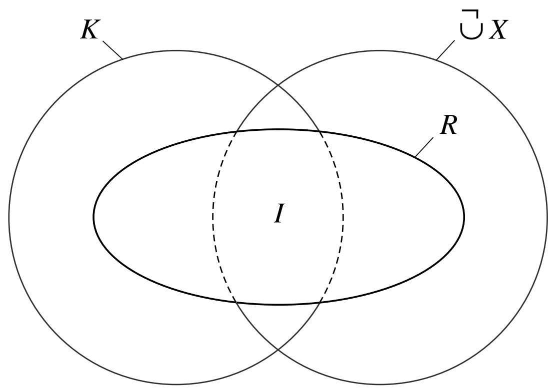


FIG. 5.1 – Multiples décompositions sur I d'une réponse R d'un agent $[K, X]$

Montrons qu'il existe une méthode générale de construction de R.

Propriété 5.2 (Réduction de l'ensemble d'interdits). *Les réponses d'un agent à un ensemble de formules restent identiques lorsqu'on ampute l'ensemble des interdits de cet agent des formules équivalentes à la négation de formules de sa base de connaissances :*

$$\mathcal{R}^A_{[K,X]} = \mathcal{R}^A_{[K,X \setminus \overset{\neg}{\cup}K]}$$

DÉMONSTRATION.

($\subseteq$) Soit R une réponse de l'agent $[K, X]$ à A. En conformité avec la figure 5.1, on note I l'intersection entre K, $\overset{\neg}{\cup}X$ et R. De par la définition d'une réponse, il existe des ensembles K' et X' tels que

$R = K' \cup (\vec{\cup} X')$, avec K' inclus dans K et X' inclus dans X. De plus :

$$\exists x \in (\perp \cup X') \setminus \overline{K'},\ K' \cup A \vdash x$$

- Si x est tel que $\neg x$ n'appartient pas à K, alors d'une part x n'appartient pas à $\vec{\cup} I$ et donc x appartient à $\perp \cup X' \setminus \vec{\cup} I$, et d'autre part, comme R est consistant et contient $\neg x$, x ne peut être inféré par $K' \cup I$. Ainsi, en posant $K'' = K' \cup I$ et $X'' = X' \setminus \vec{\cup} I$, on aboutit à :
$$\exists x \in (\perp \cup X'') \setminus \overline{K''},\ K'' \cup A \vdash x$$
- Si x est tel que $\neg x$ appartient à K, alors comme $K' \cup A \vdash x$ implique que $K' \cup \{\neg x\} \cup A \vdash \perp$, comme $\neg x$ appartient à I et comme $\perp$ appartient à $(\perp \cup X' \setminus \vec{\cup} I) \setminus \overline{K' \cup I}$ (K est consistant), on a, en posant $K'' = K' \cup I$ et $X'' = X' \setminus \vec{\cup} I$:
$$\exists x' \in (\perp \cup X'') \setminus \overline{K''},\ K'' \cup A \vdash x'$$

Or, dans un cas comme dans l'autre : $K'' \cup X'' = K' \cup I \cup \big(\vec{\cup}(X' \setminus \vec{\cup} I)\big)$ $= K' \cup I \cup (\vec{\cup} X' \setminus I) = (R \cap K) \cup (R \setminus K) = R$. De plus, K'' est inclus dans K, et X'' est inclus dans $X \setminus \vec{\cup} I$, inclus lui-même dans $X \setminus \vec{\cup} K$. Par conséquent R est une réponse de l'agent $[K, X \setminus \vec{\cup} K]$ à A.

($\supseteq$) Soit R une réponse de l'agent $[K, X \setminus \vec{\cup} K]$ à A. Alors par définition il existe des ensembles K' et X' tels que $R = K' \cup (\vec{\cup} X')$, avec K' inclus dans K et X' inclus dans $X \setminus \vec{\cup} K$. De plus :

$$\exists x \in (\perp \cup X') \setminus \overline{K'},\ K' \cup A \vdash x$$

Le fait d'ajouter des formules à l'ensemble des formules interdites de l'agent ne peut remettre en question l'existence d'un tel x. Par conséquent, R est également une réponse de l'agent $[K, X]$ à A.

$\square$

Propriété 5.3 (Reformulation d'une réponse). *Un ensemble consistant de formules R est une réponse d'un agent $[K, X]$ à un ensemble consistant de formules A si et seulement si :*

1. $R \subseteq K \cup (\vec{\cup} X)$
2. $R \cap K \nvdash_{\{\perp\} \cup X'} A$, *où* $X' = \vec{\cup}(R \setminus K)$

DÉMONSTRATION.

(⇒) Par hypothèse, R est une réponse de l'agent $[K, X]$ à un ensemble A. De par la propriété de réduction de l'ensemble d'interdits (propriété 5.2), R est donc une réponse de l'agent $[K, X \setminus \overset{\neg}{\cup} K]$ à A. Par conséquent il existe un K' inclus dans K et un X' inclus dans $X \setminus \overset{\neg}{\cup} K$ tels que : $R = K' \cup (\overset{\neg}{\cup} X')$ et $K' \nvdash_{\{\perp\} \cup X'} A$. On a donc bien : $R \subseteq K \cup (\overset{\neg}{\cup} X)$. Par ailleurs (propriété 2.4) :

$$K' \nvdash_{\{\perp\} \cup X'} A \quad \text{ssi} \quad \exists x \in (\{\perp\} \cup X') \setminus \overline{K'},\ K' \cup A \vdash x$$

Or R est consistant et contient la négation d'un tel x, car $\overset{\neg}{\cup} X'$ est inclus dans R, donc x ne peut être inféré par $R \cap K$. Alors x appartient à $(\{\perp\} \cup X') \setminus \overline{R \cap K}$, ce qui implique :

$$\exists x \in (\{\perp\} \cup X') \setminus \overline{R \cap K},\ (R \cap K) \cup A \vdash x$$

Comme X' est inclus dans $X \setminus \overset{\neg}{\cup} K$, et comme $X \setminus \overset{\neg}{\cup} K = \overset{\neg}{\cup}(R \setminus K)$:

$$\exists x \in \big(\{\perp\} \cup (\overset{\neg}{\cup}(R \setminus K))\big) \setminus \overline{R \cap K},\ (R \cap K) \cup A \vdash x$$

Par conséquent : $R \cap K \nvdash_{\{\perp\} \cup X''} A$, où $X'' = \overset{\neg}{\cup}(R \setminus K)$.

(⇐) Par hypothèse, $X' = \overset{\neg}{\cup}(R \setminus K)$. Posons $K' = R \cap K$, et $R = K' \cup (\overset{\neg}{\cup} X')$. On a bien K' inclus dans K, et comme $X' = \overset{\neg}{\cup}(R \setminus K) = X \setminus \overset{\neg}{\cup} K$ alors X' est inclus dans X. D'autre part : $R \cap K \nvdash_{\{\perp\} \cup X'} A$, donc : $K' \nvdash_{\{\perp\} \cup X'} A$. Par conséquent nous pouvons appliquer la définition d'une réponse (définition 5.1) et conclure que R est bien une réponse de $[K, X]$ à A.

□

Cette nouvelle façon de voir une réponse lève l'incertitude qui pouvait survenir lors de la décomposition d'une réponse selon les connaissances et interdits d'un agent. Bien que l'on y décompose une réponse suivant K — $R \setminus K$ et $R \cap K$ — il n'est pas question de priorité des connaissances sur les interdits, car quelle que soit la décomposition la réponse reste la même. Cette propriété fournit une méthode si l'on souhaite obtenir les K' et X' de la définition 5.1 à partir de R : on prend $K' = R \cap K$, et l'ensemble des négations des formules de $R \setminus K$ constitue X'.

De façon similaire, nous avons essayé de décomposer une réponse non plus sur les connaissances, mais sur les interdits de l'agent : pour une réponse R incluse dans $K \cup (\overset{\neg}{\cup} X)$, la décomposition tentée est $R = (R \setminus Y) \cup (R \cap Y)$

avec $Y = \overset{\neg}{\cup}X$, au lieu de $R = (R \cap K) \cup (R \setminus K)$. Mais ceci ne fonctionne pas systématiquement :

Propriété 5.4. *Il existe des agents $[K, X]$ et des ensembles de formules consistants A tels que, pour $Y = \overset{\neg}{\cup}X$ et $X' = \overset{\neg}{\cup}(R \cap Y)$:*

$$\exists R \in \mathcal{R}^A_{[K,X]}, \quad R \setminus Y \vdash_{\{\bot\}\cup X'} A$$

DÉMONSTRATION. Posons $K = \{a, b\}$, $X = \{\bot, \neg b, c\}$, $A = \{a \wedge b \Rightarrow c\}$ et $R = \{a, b, \neg c\}$. R est bien une réponse de $[K, X]$ à A, mais pour autant : $\{a\} \vdash_{\{\bot,\neg b,c\}} a \wedge b \Rightarrow c$. □

Lors de la définition de la recevabilité d'une formule par un agent (définition 3.2), nous avons souligné une différence de prise en compte importante entre connaissances et interdits d'un agent : on calcule la clôture déductive des connaissances uniquement. Les réponses reposant également sur la notion de recevabilité, cette disparité est toujours présente et les propriétés 5.3 et 5.4 ne font que le confirmer. Mettre une formule x dans l'ensemble des interdits d'un agent ne revient pas au même que d'ajouter $\neg x$ à ses connaissances.

Propriété 5.5 (Inconsistance)**.** *Tout ensemble de formules réponse d'un agent à un autre ensemble de formules est inconsistant avec ce dernier. Autrement dit :*

$$\forall \Phi, \forall A, \forall R \in \mathcal{R}^A_{\Phi}, \quad R \cup A \vdash \bot$$

La réciproque est fausse même en choisissant R inclus dans $K \cup (\overset{\neg}{\cup}X)$, avec $[K, X] = \Phi$.

DÉMONSTRATION. Soit R la réponse d'un agent $[K, X]$ à un ensemble A. D'après la définition d'une réponse : $K' \nvdash_{\{\bot\}\cup X'} A$. Donc :

$$\exists x \in (\{\bot\} \cup X') \setminus \overline{K'}, \; K' \cup A \vdash x$$

Alors pour cet x on a : $K' \cup \{\neg x\} \cup A \vdash \bot$. Ainsi : $K' \cup (\overset{\neg}{\cup}X') \cup A \vdash \bot$, et par conséquent R est inconsistant avec A.

Concernant la réciproque, prenons un agent $[\varnothing, \{\bot, a, b\}]$ et posons $A = \{a \vee b\}$. L'ensemble $R = \{\neg a, \neg b\}$ est bien inconsistant avec A, mais l'on n'a pas : $\nvdash_{\{\bot,a,b\}} A$. La condition 2 de la définition d'une réponse (définition 5.1) n'étant pas satisfaite, R n'est donc pas une réponse de cet agent à A. □

Propriété 5.6 (Non-vacuité)**.** *Une réponse ne peut être vide.*

Démonstration. Supposons que R soit une réponse vide de Φ à A. La propriété 5.5 affirmant que $R \cup A$ est inconsistant, on en déduit que A est inconsistant, ce qui est en contradiction avec la définition d'une réponse (définition 5.1). □

Propriété 5.7. *Soient R un sous-ensemble des connaissances d'un agent et A un ensemble consistant de formules, alors R est une réponse de cet agent à A si et seulement si R est inconsistant avec A :*

$$\begin{array}{l} \forall [K, X] \\ \forall A \nvdash \bot \\ \forall R \subseteq K \end{array} \quad , \quad R \in \mathcal{R}^A_{[K,X]} \ \textit{ssi} \ R \cup A \vdash \bot$$

Démonstration.

(⇒) C'est une conséquence directe de la propriété d'inconsistance des réponses (propriété 5.5).

(⇐) Cherchons à appliquer la définition d'une réponse (définition 5.1). Comme R est inclus dans K nous déduisons que R est consistant. Par ailleurs, $R \cup A \vdash \bot$ implique que R infère $\neg A$. De par la relation de consistance de la X-inférence, il en résulte que : $R \nvdash_{\{\bot\}} A$. Toutes les conditions sont alors réunies, la définition 5.1 donne à R le statut de réponse de $[K, X]$ à A.

□

Dès la construction des attitudes (*cf.* section 3.2 sur le nommage des attitudes) nous avons mentionné notre préférence pour la non-recevabilité sur la recevabilité. Au contraire de la recevabilité, la non-recevabilité garantie l'existence de formules justifiant la prise de position. Si un ensemble est non-recevable par un agent, nous sommes assurés que ce dernier a matière à argumenter. Il peut donc générer des réponses :

Propriété 5.8 (Existence d'une réponse)**.** *Si un ensemble consistant de formules n'est pas recevable par un agent alors il existe une réponse de ce dernier à cet ensemble.*

$$\forall \Phi, \ \forall A \nvdash \bot, \quad \Phi \mid\!\!\!/ \; A \neq \varnothing \ \Rightarrow \ \mathcal{R}^A_\Phi \neq \varnothing$$

La réciproque est fausse.

DÉMONSTRATION. Soit $\Phi = [K, X]$. Par hypothèse : $\exists x \in X \backslash \overline{K},\ K \cup A \vdash x$. Posons $R = K \cup \{\neg x\}$. On a bien $K \nvdash_{\{\perp,x\}} A$. Comme $[K, X]$ est un agent, K est consistant, et comme $x \notin \overline{K}$, R l'est aussi. Donc R est une réponse de $[K, X]$ à A.

Par contre, $\mathcal{R}^{A}_{[K,X]}$ non vide n'implique pas que A est non-recevable par l'agent. Par exemple : $[\{a, a \Rightarrow b\}, \{\perp, b\}]$, $A = \{a\}$ et $R = \{a \Rightarrow b, \neg b\}$. □

Comme toutes les attitudes, mise à part l'attitude neutre, contiennent dans leur définition une relation de non-recevabilité (*cf.* classification des attitudes, tableau 3.2), cette propriété sera à la base de la génération d'arguments (section 6.2).

Le fait que l'on puisse être en présence d'une réponse à un ensemble de formules sans que ce dernier soit non-recevable par l'agent concerné — comme le montre l'exemple cité dans la démonstration ci-dessus — dénote d'une grande permissivité dans la définition-même d'une réponse (définition 5.1). Elle est à notre avantage car elle permet une caractérisation d'une forme singulière de réponses, appelée *mensonge*, que nous détaillerons dans la section 5.4.

Propriété 5.9 (Inflation). *Les réponses d'un agent $[K', X']$ à un ensemble de formules A' sont des réponses de l'agent $[K, X]$ à A si K, X et A contiennent respectivement K', X' et A'. Autrement dit :*

$$\begin{array}{l} \forall K' \subseteq K \\ \forall X' \subseteq X \\ \textit{et } \forall A' \subseteq A \end{array} \quad , \quad \mathcal{R}^{A'}_{[K',X']} \subseteq \mathcal{R}^{A}_{[K,X]}$$

DÉMONSTRATION. Soit R appartenant à $\mathcal{R}^{A'}_{[K',X']}$. On pose : $X'' = \{x \in X' \mid \neg x \in R \backslash K'\}$. D'après la reformulation de la définition d'une réponse (propriété 5.3) nous pouvons écrire :

$$\exists x \in (\{\perp\} \cup X'') \setminus \overline{R \cap K'}, \quad (R \cap K') \cup A' \vdash x \tag{1}$$

Traitons séparément les cas K, X et A :

1. Pour tout K contenant K' et quel que soit x, $(R \cap K') \cup A' \vdash x$ implique $(R \cap K) \cup A' \vdash x$. Donc (1) peut se réécrire en :

$$\exists x \in (\{\perp\} \cup X'') \setminus \overline{R \cap K'}, \quad (R \cap K) \cup A' \vdash x$$

L'accroissement de K' en K peut remettre en question l'existence de x. Toutefois si $x = \perp$ il n'y aura pas de problème vu que K est consistant — c'est une base de connaissances — donc x ne peut appartenir à $\overline{R \cap K'}$. Par contre une alternative se présente si x appartient à X'', selon que $\neg x$ appartienne ou non à $K \setminus K'$. Dans le premier cas, où $\neg x \in K \setminus K'$, alors $(R \cap K') \cup A' \vdash x$ implique $(R \cap K) \cup A' \vdash \perp$, et il s'ensuit, pour $X''_2 = \{x \in X' \mid \neg x \in R \setminus K\}$, que :

$$\exists x \in (\{\perp\} \cup X''_2) \setminus \overline{R \cap K}, \quad (R \cap K) \cup A' \vdash x$$

à savoir $x = \perp$. Dans le second cas, où $\neg x \notin K \setminus K'$, alors $\neg x$ appartient à $R \setminus K$ et donc par suite à X''_2. Par ailleurs $R \cap K$ ne peut inférer x vu que R est consistant et $\neg x$ appartient à R. En conséquence, dans ce cas également :

$$\exists x \in (\{\perp\} \cup X''_2) \setminus \overline{R \cap K}, \quad (R \cap K) \cup A' \vdash x$$

à savoir le même x qu'en (1).

2. Pour tout X contenant X', si x appartient à X'' alors x appartient à $\{x \in X \mid \neg x \in R \setminus K'\}$. Donc toute réponse de $\mathcal{R}^{A'}_{[K',X']}$ est une réponse de $\mathcal{R}^{A'}_{[K',X]}$.
3. Pour tout A contenant A' et quel que soit x, $(R \cap K') \cup A' \vdash x$ implique $(R \cap K') \cup A \vdash x$. Donc (1) peut se réécrire en :

$$\exists x \in (\{\perp\} \cup X'') \setminus \overline{R \cap K'}, \quad (R \cap K') \cup A \vdash x$$

Ainsi toute réponse de $\mathcal{R}^{A'}_{[K',X']}$ est une réponse de $\mathcal{R}^{A}_{[K',X']}$.

Récapitulons : $\mathcal{R}^{A'}_{[K',X']} \subseteq \mathcal{R}^{A'}_{[K,X']} \subseteq \mathcal{R}^{A'}_{[K,X]} \subseteq \mathcal{R}^{A}_{[K,X]}$. □

Cette dernière propriété semble aller de soi : plus il y a d'informations, plus il y a de possiblités de réponses. Étant donné qu'une réponse se construit sur des connaissances et interdits par rapport à un ensemble tiers, plus ces trois ensembles seront conséquents, plus il y aura de réponses potentielles.

5.2 Réponses pertinentes

Parmi les réponses d'un agent à un ensemble de formules A, certaines sont incluses dans d'autres : elles contiennent moins d'informations superflues. À l'extrême, les réponses minimales au sens de l'inclusion ne sont alors plus constituées que de formules nécessaires à l'élaboration de la conclusion $\neg A$. Nous qualifions ces réponses de *pertinentes*.

Définition 5.10 (Réponse pertinente)**.** *Une réponse de l'agent Φ à l'ensemble de formules A est dite* pertinente *si et seulement si elle ne contient aucune autre réponse de Φ à A. On note $\mathcal{R}p^A_\Phi$ l'ensemble des* réponses pertinentes *de Φ à A.*

$$\mathcal{R}p^A_\Phi \;=\; \left\{R \in \mathcal{R}^A_\Phi \mid \forall R' \subset R, R' \notin \mathcal{R}^A_\Phi\right\}$$

Cette contrainte de minimalité a des répercussions sur le nombre d'interdits appartenant à l'agent qu'une réponse pertinente peut encapsuler :

Propriété 5.11 (Interdits et pertinence)**.** *La partie d'une réponse pertinente d'un agent non incluse dans les connaissances de ce dernier est soit vide soit un singleton contenant la négation de l'un des interdits consistants de l'agent :*

$$\forall R \in \mathcal{R}p^A_{[K,X]}, \quad R \setminus K \in \left\{\{\neg x\} \;\mid\; x \in X \text{ et } \{x\} \nvdash \bot\right\} \cup \varnothing$$

DÉMONSTRATION. Soit R une réponse pertinente de $[K, X]$ à A telle que $\mathsf{card}\,(R \setminus K) > 1$. De par la propriété de reformulation d'une réponse (propriété 5.3) on a :

$$\exists x \in \left(\{\bot\} \cup (\bar{\bigcup}(R \setminus K))\right) \setminus \overline{R \cap K}, \quad (R \cap K) \cup A \vdash x$$

Soit $x = \bot$ soit x appartient à $\bar{\bigcup}(R \setminus K)$. Dans les deux cas $R \cap K$ est une réponse de $[K, X]$ à A strictement incluse dans R Ceci contredit la pertinence de la réponse R. Par conséquent le cardinal de $R \setminus K$ est inférieur ou égal à 1.

Comme dans la définition d'une réponse, la contradiction est systématiquement ajoutée à l'ensemble X de la X-inférence, les réponses R utilisant un interdit inconsistant de l'agent ne peuvent être des réponses pertinentes : en effet, $R \setminus \{\top\}$ est aussi une réponse. □

Cette propriété provient de l'utilisation d'une X-inférence lors de la définition de ce qu'est une réponse. Plus précisément, étant donné que la mise en cause de plusieurs interdits dans l'échec du processus de X-inférence d'une formule ne peut être avérée qu'à la condition que chacun de ses interdits y soit un obstacle, une réponse ne peut solliciter plusieurs interdits sans pouvoir se satisfaire d'un parmi ceux-ci. Cela explique pourquoi les réponses pertinentes n'en contiennent donc qu'un au plus.

Propriété 5.12. *Tout ensemble de formules strictement inclus dans une réponse pertinente d'un agent à un autre ensemble de formules est consistant avec ce dernier.*

$$\forall R \in \mathcal{R}p_{\Phi}^{A}, \quad R' \subset R \Rightarrow R' \cup A \nvdash \bot$$

Démonstration. Posons $\Phi = [K, X]$. Supposons qu'il existe une réponse pertinente R de Φ à A et un ensemble R', strictement contenu dans R, inconsistant avec A. Considérons deux options, suivant que R' est inclus ou non dans K :

($R' \subseteq K$) L'équivalence de la propriété 5.7 indique que R' est une réponse de Φ à A. Mais : $R \in \mathcal{R}_{\Phi}^{A}$, alors : $\forall R'' \in \mathcal{R}_{\Phi}^{A}$, $R'' \not\subset R$. Comme R' est strictement inclus dans R, il y a contradiction.

($R' \nsubseteq K$) D'après la propriété 5.11, R' est donc l'union de $R' \setminus K$ et d'un singleton contenant la négation de l'un des interdits consistants x de X. Alors $R' \cup A \vdash \bot$ implique $(R' \cap K) \cup \{\neg x\} \cup A \vdash \bot$. Donc : $(R' \cap K) \cup A \vdash x$. Comme $\neg x$ appartient à R', $R' \cap K$ ne peut inférer x sans remettre en cause la consistance de R'. Ainsi :

$$\exists x \in \big(\{\bot\} \cup (\bar{\bigcup}(R' \setminus K))\big) \setminus \overline{R' \cap K}, \quad (R' \cap K) \cup A \vdash x$$

Les conditions de la seconde formulation de la définition d'une réponse sont réunies (propriété 5.3), R' est une réponse de Φ à A. Cela contredit le fait que R est une réponse pertinente.

Il y a contradiction dans les deux cas, par conséquent si R' est strictement inclus dans R alors R' est consistant avec A. □

5.3 Génération de réponses pertinentes

Prenons un agent $[K, X]$ confronté à un ensemble de formules A. Son ensemble de réponses pertinentes peut être partitionné en trois catégories. La première contient les réponses établies uniquement à partir des connaissances de l'agent. Cette partition est donc non vide dès lors que K et A sont inconsistants entre eux. Une deuxième catégorie de réponses est constituée de toutes celles construites exclusivement à partir de l'ensemble des formules interdites de l'agent. De la propriété traitant du rapport entre interdits et pertinence (propriété 5.11) découle le fait que seules des réponses ne contenant qu'une unique formule appartiennent à cette partition. Enfin, l'ensemble des réponses pertinentes ne se trouvant ni dans la première ni dans la seconde partition constitue la troisième catégorie. Elles contiennent de fait au moins une formule de l'ensemble des connaissances de l'agent, ainsi qu'au moins un interdit de ce dernier.

Ce partitionnement peut être vu au travers d'agents « virtuels », c'est-à-dire des agents que l'on crée de toute pièce. Les réponses élaborées uniquement à partir des connaissances de l'agent $[K, X]$ sont en fait les réponses d'un agent $[K, \{\bot\}]$, tandis que les réponses fondées sur les seuls interdits de l'agent $[K, X]$ sont les réponses de l'agent $[\varnothing, X]$. La figure 5.2 représente un tel partitionnement.

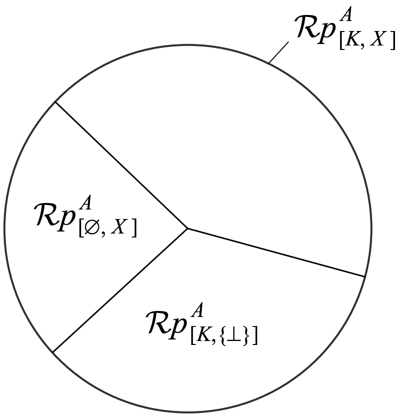


FIG. 5.2 – Partitionnement des réponses pertinentes d'un agent $[K, X]$ à un ensemble de formules A

Les deux propriétés suivantes (propriétés 5.13 et 5.14) caractérisent les ensembles de réponses pertinentes des agents $[\varnothing, X]$ et $[K, \{\bot\}]$.

Propriété 5.13 (Pertinence sans connaissances)**.** *L'ensemble des réponses pertinentes de l'agent $[\varnothing, X]$ à l'ensemble A est l'ensemble des singletons consistants non-recevables par l'agent $[A, \{\bot\}]$ et constitués de la négation d'une formule de X. Autrement dit :*

$$\mathcal{R}p^{A}_{[\varnothing,X]} = \big\{\{f\} \mid f \textit{ est consistante et } \{f\} \in [A, \{\bot\}] \mid_{\nvdash} \overset{\neg}{\cup} X\big\}$$

DÉMONSTRATION.

(⊆) Soit R une réponse pertinente de $[\varnothing, X]$ à A. D'après la propriété 5.11 sur les interdits et pertinence, $R = \{\neg x_1\}$, où x_1 est une contrainte de X. La propriété d'inconsistance des réponses (propriété 5.5) indique que $R \cup A \vdash \bot$. Donc $A \vdash x_1$, ce qui peut se réécrire, via la relation de consistance de la X-inférence, en :

$A \nvdash_{\{\bot\}} \neg x_1$. Si l'on ajoute à ceci qu'une réponse est consistante, alors :

$$\{\neg x_1\} \in \big\{\{f\} \mid f \text{ est consistante et } \{f\} \in [A, \{\bot\}] \mid_{\not\vdash} \bar{\cup} X\big\}$$

(⊇) Quel que soit $\{f\}$ appartenant à $[A, \{\bot\}] \mid_{\not\vdash} \bar{\cup} X$, $A \nvdash_{\{\bot\}} f$. Soit x_1 appartenant à X tel que $\neg x_1 = f$. Donc $A \nvdash_{\{\bot\}} \neg x_1$, puis $A \vdash x_1$. Comme f est consistante, x_1 ne peut être une tautologie et par conséquent :

$$\exists x \in \{\bot, x_1\} \setminus \overline{\top},\ A \vdash x$$

Ainsi : $\nvdash_{\{\bot, x_1\}} A$, et donc $\{f\}$ est une réponse de $[\varnothing, X]$ à A. De par la non-vacuité d'une réponse (propriété 5.6) on en conclut que $\{f\}$ est une réponse pertinente de $[\varnothing, X]$ à A.

□

Propriété 5.14 (Pertinence sans interdits)**.** *L'ensemble des réponses pertinentes de l'agent* $[K, \{\bot\}]$ *à* A *est l'ensemble des parties minimales de* K *non-recevables par l'agent* $[A, \{\bot\}]$.

$$\mathcal{R}p^{A}_{[K,\{\bot\}]} = \mathsf{min}\big([A, \{\bot\}] \mid_{\not\vdash} K\big)$$

Démonstration.

(⊇) Soit R appartenant à $\mathsf{min}\big([A, \{\bot\}] \mid_{\not\vdash} K\big)$. Donc $R \subseteq K$ et $A \nvdash_{\{\bot\}} R$. Via la relation de consistance de la X-inférence, on a : $A \vdash \neg R$, et par suite : $R \cup A \vdash \bot$. Comme $[A, \{\bot\}]$ est un agent, A est consistant. Nous pouvons donc appliquer la propriété 5.7 et obtenir que R est une réponse de $[K, \{\bot\}]$ à A. Nous avons choisi R appartenant à l'ensemble des minimaux de $[A, \{\bot\}] \mid_{\not\vdash} K$. Ainsi il ne peut exister de R' strictement inclus dans R tel que R' appartienne à $\mathcal{R}^{A}_{[K,\{\bot\}]}$. Par conséquent, R est une réponse pertinente de $[K, \{\bot\}]$ à A.

(⊆) Soit R appartenant aux réponses pertinentes de $[K, \{\bot\}]$ à A. Donc $R \subseteq K$ et $R \cup A$ est inconsistant. Alors $A \vdash \neg R$, c'est-à-dire $A \nvdash_{\{\bot\}} R$. Ainsi R appartient à $[A, \{\bot\}] \mid_{\not\vdash} K$. Ci-dessus nous avons montré que s'il existe un R' strictement inclus dans R et appartenant à $[A, \{\bot\}] \mid_{\not\vdash} K$, alors R' est une réponse de $[K, \{\bot\}]$ à A, en contradiction avec le fait que R est une réponse pertinente de $[K, \{\bot\}]$ à A. Par conséquent il n'existe pas de R' strictement inclus dans R et appartenant à $[A, \{\bot\}] \mid_{\not\vdash} K$ et donc R est l'un des minimaux de $[A, \{\bot\}] \mid_{\not\vdash} K$.

□

On remarquera que les réponses pertinentes de l'agent $[K, \{\bot\}]$ à A sont les sous-ensembles minimaux de K inconsistants avec A.

5.4 Mensonge

Nous avons montré dans [AR06] que les réponses sont ainsi définies (définition 5.1) qu'il est possible pour un agent d'en élaborer de telles qu'elles soient inconsistantes avec ses propres connaissances. Nous nommons ces réponses des *mensonges* :

Définition 5.15 (Mensonge). *M est un* mensonge *de l'agent $[K, X]$ à propos de l'ensemble de formules A si et seulement si :*

1. *M est une réponse pertinente de $[K, X]$ à A ;*
2. *M est inconsistant avec K.*

L'ensemble des mensonges de $[K, X]$ à propos de A est noté $\mathcal{R}m^{A}_{[K,X]}$.

Cette capacité relève de la façon dont est maniée la X-inférence au sein de la définition même d'une réponse : lors de son élaboration, il est autorisé pour un agent $[K, X]$ de requérir une formule de $X \cap \overline{K}$. Rappelons que suivant la propriété de la X-inférence (propriété 2.4) :

$$K \nvdash_X A \quad \text{ssi} \quad \exists x \in X \setminus \overline{K},\ K \cup A \vdash x$$

Or une réponse $R = K' \cup (\bar{\cup} X')$ de cet agent à A vérifiera seulement que $K' \nvdash_{\{\bot\} \cup X'} A$, c'est-à-dire :

$$\exists x \in (\{\bot\} \cup X') \setminus \overline{K'},\ K' \cup A \vdash x$$

Une formule de $X \cap \overline{K} \setminus K'$ peut alors être la cause de la non-inférence et expliquer l'inconsistance de la réponse avec les connaissances de l'agent.

Propriété 5.16 (Mensonge et interdits). *Si M est une réponse pertinente de l'agent $[K, X]$ à l'ensemble de formules A, alors M est un mensonge de cet agent à propos de A si et seulement si il existe une formule de $M \setminus K$ inconsistante avec K. Autrement dit :*

$$\forall M \in \mathcal{R}p^{A}_{[K,X]}, \quad M \in \mathcal{R}m^{A}_{[K,X]} \ \textit{ssi} \ \exists m \in M \setminus K,\ K \cup \{m\} \vdash \bot$$

DÉMONSTRATION.

($\Rightarrow$) M est un mensonge, donc $M \cup K$ est inconsistant. Alors : $(M \cap K) \cup (M \setminus K) \cup K \vdash \bot$, ce qui implique que : $(M \setminus K) \cup K \vdash \bot$. La propriété interdits et pertinence (propriété 5.11) indique que $M \setminus K$ est soit vide, soit un singleton. Comme K est une base de connaissances et est donc consistant, $M \setminus K$ ne peut être vide. Par conséquent il existe une formule de $M \setminus K$ inconsistante avec K.

($\Leftarrow$) Par hypothèse, $K \cup \{m\}$ est inconsistant. Donc $K \cup M$ l'est également. Comme M est une réponse pertinente de $[K, X]$ à A, la définition d'un mensonge (définition 5.15) précise que M est un mensonge de $[K, X]$ à propos de A.

$\square$

Cette propriété laisse sous-entendre qu'une relation entre connaissances et interdits d'un agent est nécessaire à la formation de mensonges :

Corollaire 5.17. *Pour tout agent $[K, X]$ et pour tout ensemble de formules A, si l'intersection entre X et les théorèmes de K est vide, alors l'agent ne peut formuler aucun mensonge à propos de A :*

$$\forall [K, X], \forall A, \quad X \cap \overline{K} = \varnothing \quad \Rightarrow \quad \mathcal{R}m^{A}_{[K,X]} = \varnothing$$

Démonstration. Supposons que l'agent $[K, X]$ puisse construire un mensonge M à propos de A. De la propriété mensonge et interdits (propriété 5.16) découle qu'il existe une formule m appartenant à $M \setminus K$ telle que $K \vdash \neg m$. Or, par définition d'une réponse (définition 5.1), $M \setminus K$ est inclus dans l'ensemble des négations des formules de X. Donc il existe une formule de X inférée par K et par conséquent : $X \cap \overline{K} \neq \varnothing$. Il ne nous reste qu'a prendre la contre-apposée. $\square$

Ce corollaire amène à conclure qu'un agent $[\varnothing, X]$ ou $[K, \{\bot\}]$, quels que soient K et X, ne peut générer de mensonge. À noter cependant qu'une réponse pertinente de l'agent $[\varnothing, X]$ à un ensemble A peut devenir un mensonge à propos de A pour un agent de mêmes interdits mais avec des connaissances appropriées :

Exemple 5.18. Soient deux agents Φ et Ψ tels que : $\Phi = [\varnothing, \{\bot, a\}]$, et : $\Psi = [\{a\}, \{\bot, a\}]$. Pour un ensemble $A = \{a\}$, nous obtenons :
- $\mathcal{R}m^{A}_{\Phi} = \varnothing$ et $\mathcal{R}p^{A}_{\Phi} = \big\{\{\neg a\}\big\}$
- $\mathcal{R}m^{A}_{\Psi} = \big\{\{\neg a\}\big\}$

$\diamondsuit$

La figure 5.2 faisant état d'un partitionnement des réponses pertinentes, elle peut être réutilisée afin d'exhiber une localisation des mensonges potentiels d'un agent $[K, X]$ à propos d'un ensemble de formules A. Nous aboutissons à la figure 5.3.

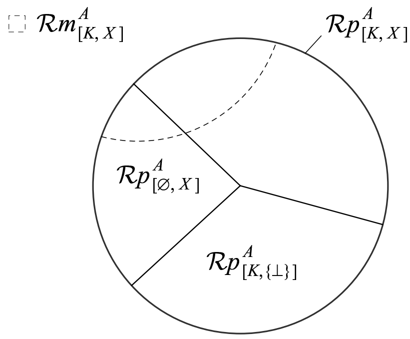


FIG. 5.3 – Localisation des mensonges d'un agent $[K, X]$ à propos d'un ensemble de formules A

La section 6.3 est une illustration qui mélange génération d'arguments et utilisation du mensonge.

Chapitre 6

Arguments

Au cours de notre étude de la notion d'argument (section 1.2) nous avons recensé trois approches principales : par arbre d'inférence, par séquence d'inférences ou par paire support-conclusion. Cette dernière, illustrée par la définition 1.6 et utilisée notamment par [SL92], [EGKF93], [AC98] ou encore [BH01], nous semble la plus simple et naturellement bien adaptée pour une future comparaison de nos travaux avec une partie de ceux de Besnard et Hunter (chapitre 7). Néanmoins, si les définitions exposées dans les travaux que nous venons de citer se ressemblent beaucoup, la nôtre se présente autrement du fait de l'incorporation des notions d'agent et de réponse :

Définition 6.1 (Argument d'un agent)**.** *Un argument d'un agent Φ est un couple $\langle R, \neg A\rangle$ tel que R est une réponse de Φ à A. On note $\mathcal{A}rg \subseteq 2^{\mathcal{L}} \times \mathcal{L}$ l'ensemble des arguments, et $\mathcal{A}rg_\Phi$ l'ensemble des arguments de l'agent Φ.*

$$\forall \Phi, \forall R, \forall A, \quad \langle R, \neg A\rangle \in \mathcal{A}rg_\Phi \text{ ssi } R \in \mathcal{R}_\Phi^A$$

On note $\mathsf{supp}\,(\alpha)$ *le support d'un argument* α*, et* $\mathsf{concl}\,(\alpha)$ *sa conclusion.*

6.1 Relations inter-arguments

Les relations entre arguments sont une composante essentielle à tout système argumentatif puisqu'elles fournissent un moyen d'apprécier la *force* ou la *qualité* d'un argument, ce qui permet par exemple de décider à terme si la thèse initiale d'un débat est acceptable ou non.

Dans le but de préciser certaines relations inter-arguments, nous avons besoin de détailler le support d'un argument en y distinguant le *support minimal* :

Définition 6.2 (Support minimal). *Le support d'un argument* α *contient une partie dite* minimale *et notée* $\mathsf{suppMin}(\alpha)$ *telle que :*

$$\mathsf{suppMin}(\alpha) = \{f \in \mathsf{supp}(\alpha) \mid \forall E \subset \mathsf{suppMin}(\alpha),\ E \nvdash \mathsf{concl}(\alpha)\}$$

Définition 6.3 (Relations inter-arguments). *Soient* α *et* β *deux arguments quelconques. Nous dirons que :*

a. α attaque β *ssi* $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$,
où $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta)$ *;*

b. α *s'*oppose *à* β *ssi* $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$,
où $\{s_1, \ldots, s_n\} \subseteq \mathsf{suppMin}(\beta)$ *;*

c. α contrarie β *ssi* $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$,
où $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta) \setminus \mathsf{suppMin}(\beta)$ *;*

d. α réfute β *ssi* $\mathsf{concl}(\alpha) = \neg\mathsf{concl}(\beta)$ *;*

e. α défend β *ssi* $\mathsf{concl}(\alpha) = s_1 \wedge \cdots \wedge s_n$*, où* $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta)$ *;*

f. α approuve β *ssi* $\mathsf{concl}(\alpha) = \mathsf{concl}(\beta)$.

On note $\mathcal{A}rg_\Phi^{\mathsf{att}(\alpha)}$, $\mathcal{A}rg_\Phi^{\mathsf{cont}(\alpha)}$, $\mathcal{A}rg_\Phi^{\mathsf{réf}(\alpha)}$, $\mathcal{A}rg_\Phi^{\mathsf{opp}(\alpha)}$, $\mathcal{A}rg_\Phi^{\mathsf{déf}(\alpha)}$ *et* $\mathcal{A}rg_\Phi^{\mathsf{app}(\alpha)}$ *les ensembles d'arguments de l'agent* Φ *qui respectivement attaquent, contrarient, réfutent, s'opposent à, défendent et approuvent l'argument* α *:*

$$\begin{array}{lcl}
\mathcal{A}rg_\Phi^{\mathsf{att}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ attaque } \alpha\} \\
\mathcal{A}rg_\Phi^{\mathsf{cont}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ contrarie } \alpha\} \\
\mathcal{A}rg_\Phi^{\mathsf{réf}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ réfute } \alpha\} \\
\mathcal{A}rg_\Phi^{\mathsf{opp}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ s'oppose à } \alpha\} \\
\mathcal{A}rg_\Phi^{\mathsf{déf}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ défend } \alpha\} \\
\mathcal{A}rg_\Phi^{\mathsf{app}(\alpha)} & = & \{\beta \in \mathcal{A}rg_\Phi \mid \beta \text{ approuve } \alpha\}
\end{array}$$

Remarque 6.4 (Relations et conflits). Nous faisons état de *relations* et non de *conflits* entre arguments. Si de prime abord la nuance est subtile, elle devient sensible lorsque l'on s'intéresse à la défense d'un argument. En effet cette relation est généralement abordée au travers de l'attaque, quand elle n'est pas éludée.

Prenons par exemple la définition de la *défense conjointe* introduite dans [Dun95]. Il s'agit en fait de la définition 1.15 dans laquelle nous avons traduit « défendu par » par « acceptable pour ». Un argument α y est dit défendable par un ensemble d'arguments S si et seulement si quel que soit l'argument β s'opposant à α alors il existe un argument de S s'opposant à β. Par conséquent un argument non défendable à cause d'un unique β non contre-attaqué n'appartiendra pas à cette classe d'acceptabilité quand bien même nombre d'arguments viendraient le défendre — ou l'approuver — au sens de notre définition 6.3.

Une autre illustration peut être vue en la notion de *défense individuelle* présentée dans [AC98] (définition 1.12) : pour qu'un argument α puisse se défendre seul contre un argument β il faut entre autre que β s'oppose à α.

Nous montrons dans la section suivante (section 6.2) comment générer des arguments de défense par le biais de nos attitudes et de la notion de réponse. $\diamondsuit$

De la définition de relations entre arguments découlent différentes propriétés :

Propriété 6.5. *Un argument ne peut simultanément attaquer et défendre un autre argument :*

$$\forall \Phi, \forall \alpha, \quad \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)} \cap \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)} = \varnothing$$

Démonstration. Si un argument β attaque et défend simultanément un argument α alors : $\mathsf{concl}\,(\beta) = s_i \wedge \cdots \wedge s_j = \neg(s_k \wedge \cdots \wedge s_l)$, avec : $\{s_i, \ldots, s_j\} \subseteq \mathsf{supp}\,(\alpha)$, et : $\{s_k, \ldots, s_l\} \subseteq \mathsf{supp}\,(\alpha)$. Donc : $\{s_i, \ldots, s_j\} \vdash \neg(s_k \wedge \cdots \wedge s_l)$, et l'on obtient : $\mathsf{supp}\,(\alpha) \vdash \bot$, ce qui est en contradiction avec le fait qu'une réponse — le support d'un argument — est consistante. Ainsi un argument ne peut à la fois attaquer et défendre un autre argument. $\square$

Propriété 6.6. *Tout argument s'opposant à un autre argument est aussi un attaquant de ce dernier :*

$$\forall \Phi, \forall \alpha, \quad \mathcal{A}rg_{\Phi}^{\mathsf{opp}(\alpha)} \subseteq \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$$

DÉMONSTRATION. Si β s'oppose à α alors : $\mathsf{concl}(\beta) = \neg(s_1 \wedge \cdots \wedge s_n)$, avec $\{s_1, \ldots, s_n\} \subseteq \mathsf{suppMin}(\alpha)$. Or : $\mathsf{suppMin}(\alpha) \subseteq \mathsf{supp}(\alpha)$ (*cf.* définition 6.2 sur le support minimal). Par conséquent si β s'oppose à α alors β attaque α. $\square$

Propriété 6.7. *Tout argument contrariant un autre argument est aussi un attaquant de ce dernier :*

$$\forall\Phi, \forall\alpha, \quad \mathcal{A}rg_{\Phi}^{\mathsf{cont}(\alpha)} \subseteq \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$$

DÉMONSTRATION. Si α contrarie β alors $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$, avec $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta) \setminus \mathsf{suppMin}(\beta)$. Donc $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta)$. Par conséquent α attaque β. $\square$

Propriété 6.8. *Un argument ne peut simultanément contrarier et s'opposer à un autre argument :*

$$\forall\Phi, \forall\alpha, \quad \mathcal{A}rg_{\Phi}^{\mathsf{cont}(\alpha)} \cap \mathcal{A}rg_{\Phi}^{\mathsf{opp}(\alpha)} = \varnothing$$

DÉMONSTRATION. Si α contrarie β alors $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$, avec $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}(\beta) \setminus \mathsf{suppMin}(\beta)$. Donc $\{s_1, \ldots, s_n\} \not\subseteq \mathsf{suppMin}(\beta)$ et ainsi α ne s'oppose pas à β. Si maintenant α s'oppose à β alors $\mathsf{concl}(\alpha) = \neg(s_1 \wedge \cdots \wedge s_n)$, avec $\{s_1, \ldots, s_n\} \subseteq \mathsf{suppMin}(\beta)$. Donc $\{s_1, \ldots, s_n\} \not\subseteq \mathsf{supp}(\beta) \setminus \mathsf{suppMin}(\beta)$ et ainsi α ne contrarie pas β. $\square$

Propriété 6.9. *Si un argument β de l'agent Φ réfute un argument α, alors $\langle \mathsf{supp}(\beta), \neg\mathsf{suppMin}(\alpha)\rangle$ est un argument de Φ s'opposant à α.*

$$\forall\Phi, \forall\alpha, \forall\beta \in \mathcal{A}rg_{\Phi}^{\mathsf{réf}(\alpha)}, \quad \langle \mathsf{supp}(\beta), \neg\mathsf{suppMin}(\alpha)\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{opp}(\alpha)}$$

DÉMONSTRATION. De par la définition d'un argument, le support de l'argument α est consistant et $\mathsf{suppMin}(\alpha) \vdash \mathsf{concl}(\alpha)$. Donc $\neg\mathsf{concl}(\alpha) \vdash \neg\mathsf{suppMin}(\alpha)$. Comme $\mathsf{supp}(\beta) \vdash \mathsf{concl}(\beta)$ et que $\mathsf{concl}(\beta) = \neg\mathsf{concl}(\alpha)$, un argument de Φ s'opposant à α est : $\langle \mathsf{supp}(\beta), \neg\mathsf{suppMin}(\alpha)\rangle$. $\square$

6.2 Génération d'arguments

À présent nous allons relier les attitudes d'un agent à l'existence d'arguments lui appartenant.

La discussion (section 3.2) sur le nommage des attitudes fait part de notre préférence à la non-recevabilité sur la recevabilité, car elle nous garantit l'existence de formules de l'agent expliquant pourquoi un ensemble n'est pas recevable. C'est en s'appuyant sur ce fait que la propriété 5.8 montre l'existence d'une réponse pour un agent à un ensemble de formules dès lors que ce dernier est non-recevable par l'agent. Les réponses ayant été élaborées dans le but de construire des arguments, nous proposons d'ores et déjà de relier la non-recevabilité à l'existence d'arguments.

Mais nous insistons sur le fait suivant, cette non-recevabilité ne nous limite nullement à la génération d'arguments conflictuels : l'attitude contre et l'attitude pour sont duales, tandis que la perplexité d'un agent sur un ensemble de formules E engage la non-recevabilité simultanée de E et de $\neg E$ par cet agent (voir propriété 3.8).

Lemme 6.10 (Existence d'un argument)**.** *Si un ensemble consistant A de formules est non-recevable par un agent alors il existe un argument de cet agent dont la conclusion est $\neg A$:*

$$\forall \Phi, \forall A \nvdash \bot, \quad \Phi \mid_{\not r} A \;\Rightarrow\; \exists \langle R, \neg A \rangle \in \mathcal{A}rg_{\Phi}$$

La réciproque est fausse.

Démonstration. L'application directe de la propriété montrant l'existence d'un réponse (propriété 5.8) puis de la définition d'un argument d'un agent (définition 6.1) permet de déduire que l'argument $\langle R, \neg A \rangle$ est un argument de l'agent Φ.

Comme la réciproque de la propriété 5.8 est fausse, la réciproque de ce lemme l'est aussi. □

Propriété 6.11. *Si A est un sous-ensemble du support d'un argument α non-recevable par l'agent Φ, alors il existe un argument $\langle R, \neg A \rangle$ de Φ attaquant α :*

$$\forall \Phi, \forall A, \forall \alpha, \quad A \in \Phi \mid_{\not r} \mathsf{supp}\,(\alpha) \;\Rightarrow\; \exists \langle R, \neg A \rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$$

La réciproque est fausse.

DÉMONSTRATION. Soit : $\Phi = [K, X]$. Par hypothèse, A est inclus dans le support de α et est donc consistant, et A est non-recevable par K au regard de X. Ainsi, d'après le lemme sur l'existence d'un argument (lemme 6.10) il existe un argument $\langle R, \neg A\rangle$ de Φ qui attaque α.

Comme la réciproque du lemme 6.10 est fausse, la réciproque de cette propriété l'est aussi. $\square$

Propriété 6.12. *Si un agent est contre un sous-ensemble du support d'un argument α, alors cet agent possède au moins un argument attaquant α :*

$$\forall \Phi, \forall A, \forall \alpha, \quad A \in \Phi \mid_{-} \mathsf{supp}\,(\alpha) \;\Rightarrow\; \exists \langle R, \neg A\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$$

La réciproque est fausse.

DÉMONSTRATION. Comme $\Phi \mid_{-} \alpha$ est inclus dans $\Phi \mid_{\not{+}} \alpha$ (*cf.* propriété 4.5), le propriété 6.11 nous permet de conclure immédiatement.

La réciproque de la propriété 6.11 étant fausse, la réciproque de celle-ci l'est aussi. $\square$

Propriété 6.13. *Si un agent est pour un sous-ensemble du support d'un argument α alors cet agent possède au moins un argument défendant α :*

$$\forall \Phi, \forall A, \forall \alpha, \quad A \in \Phi \mid_{+} \mathsf{supp}\,(\alpha) \;\Rightarrow\; \exists \langle R, \bigwedge_{a \in A} a\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)}$$

La réciproque est fausse.

DÉMONSTRATION. L'agent Φ est pour A, donc $\neg A$ est non-recevable par Φ. Le lemme 6.10 nous permet d'en déduire l'existence d'un argument $\langle R, \bigwedge_{a \in A} a\rangle$ appartenant à Φ. Comme A est un sous-ensemble du support de α, $\langle R, \bigwedge_{a \in A} a\rangle$ défend α.

La réciproque du lemme 6.10 est fausse, donc la réciproque de cette propriété l'est aussi. $\square$

Propriété 6.14. *Si un agent est perplexe sur un sous-ensemble du support d'un argument α alors cet agent possède simultanément au moins un argument attaquant α et un autre le défendant :*

$$\forall \Phi, \forall A, \forall \alpha, \quad A \in \Phi \mid_{p} \mathsf{supp}\,(\alpha) \;\Rightarrow\; \begin{cases} \exists \langle R_1, \neg A\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)} \\ \exists \langle R_2, \bigwedge_{a \in A} a\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)} \end{cases}$$

La réciproque est fausse.

Démonstration. D'une part, si Φ est perlexe sur A, alors A n'est pas recevable par Φ et la propriété 6.11 nous permet d'affirmer qu'il existe un argument $\langle R_1, \neg A\rangle$ de Φ attaquant α. D'autre part, on peut aussi déduire de Φ perlexe sur A que $\neg A$ n'est pas recevable par Φ. Le lemme 6.10 sur l'existence d'un argument nous démontre alors que l'agent Φ possède un argument $\langle R, \bigwedge_{a\in A} a\rangle$. A est un sous-ensemble du support de α, par conséquent $\langle R, \bigwedge_{a\in A} a\rangle$ est un argument de Φ défendant α.

Comme les réciproques de la propriété 6.11 et du lemme 6.10 sont fausses, la réciproque de cette propriété l'est aussi. $\square$

La neutralité est l'attitude dans laquelle se trouve un agent déclarant simultanément recevables une formule et sa négation. Le fait pour un agent d'être neutre au sujet d'un sous-ensemble du support d'un argument signifie donc qu'il n'interagit pas avec ce dernier. Par conséquent ceci ne permet pas à l'agent de générer de réponse, et par suite ne lui permet pas non plus de produire un quelconque argument. La neutralité est ici synonyme de mutisme.

Propriété 6.15. *Si M est un mensonge d'un agent $[K, X]$ à propos d'un ensemble de formules A, alors cet agent possède au moins un argument $\langle R, \neg M\rangle$ attaquant l'argument $\langle M, \neg A\rangle$:*

$$\forall M \in \mathcal{Rm}^{A}_{[K,X]}, \quad \mathcal{A}rg^{\mathsf{att}(\langle M,\neg A\rangle)}_{[K,X]} \neq \varnothing$$

Démonstration. La propriété mensonge et interdits (propriété 5.16) indique qu'il existe une formule m de M telle que $K \cup \{m\}$ est inconsistant. La propriété 5.7 permet alors de considérer K comme une réponse de l'agent $[K, X]$ à $\{m\}$. En appliquant la propriété d'inflation des ensembles de réponses (propriété 5.9), on déduit que K est une réponse de $[K, X]$ à M. Ainsi l'argument $\langle K, \neg M\rangle$ est un argument de $[K, X]$ attaquant $\langle M, \neg A\rangle$, et par conséquent l'ensemble des arguments de $[K, X]$ attaquants $\langle M, \neg A\rangle$ n'est pas vide. $\square$

Afin de faciliter l'exploitation des propriétés 6.11 à 6.15, nous proposons une reformulation des ensembles des arguments d'un agent qui attaquent ou défendent un autre argument :

Propriété 6.16.

$$\begin{aligned} \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)} &= \{\langle R, \neg A\rangle \mid R \in \mathcal{R}_{\Phi}^{A} \;\; et \;\; A \subseteq \mathsf{supp}\,(\alpha)\} \\ \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)} &= \{\langle R, \bigwedge_{a\in A} a\rangle \mid R \in \mathcal{R}_{\Phi}^{\{\neg A\}} \;\; et \;\; A \subseteq \mathsf{supp}\,(\alpha)\} \end{aligned}$$

DÉMONSTRATION. La définition 6.3 sur les relations inter-arguments nous apprend, étant donné $\{s_1, \ldots, s_n\} \subseteq \mathsf{supp}\,(\alpha)$, que :

$$\begin{aligned} \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)} &= \{\beta \in \mathcal{A}rg_{\Phi} \mid \mathsf{concl}\,(\beta) = \neg(s_1 \wedge \cdots \wedge s_n)\} \\ \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)} &= \{\beta \in \mathcal{A}rg_{\Phi} \mid \mathsf{concl}\,(\beta) = s_1 \wedge \cdots \wedge s_n\} \end{aligned}$$

Comme la définition d'un argument d'un agent (définition 6.1) stipule que β est un argument de Φ si et seulement si $\mathsf{supp}\,(\beta) \in \mathcal{R}_{\Phi}^{\neg\mathsf{concl}(\beta)}$, nous aboutissons aux formulations des ensembles $\mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$ et $\mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)}$ mentionnés dans la propriété en posant respectivement $\beta = \langle R, \neg A\rangle$ et $\beta = \langle R, \bigwedge_{a\in A} a\rangle$. □

6.3 Illustration

Prenons un agent Φ dont les connaissances sont $\{linux, (linux \vee windows) \Rightarrow stable\}$ et avec pour unique formule dans son ensemble d'interdits la contradiction. Un agent Ψ avance l'argument suivant :

$$\alpha = \langle\{windows, windows \Rightarrow stable\}, stable\rangle$$

Analyse de l'argument : Φ analyse l'argument de son interlocuteur en calculant son attitude vis-à-vis des différentes parties du support de cet argument via les opérateurs de confrontation :

- Φ est neutre au sujet de $\{windows\}$,
- Φ est pour $\{windows \Rightarrow stable\}$,
- Φ est également neutre au sujet de $\{windows, windows \Rightarrow stable\}$,

ou dit autrement :

- $\Phi \mid_{+} \mathsf{supp}\,(\alpha) = \big\{\{windows \Rightarrow stable\}\big\}$
- $\Phi \mid_{0} \mathsf{supp}\,(\alpha) = \big\{\{windows\}, \{windows, windows \Rightarrow stable\}\big\}$
- $\Phi \mid_{-} \mathsf{supp}\,(\alpha) = \varnothing$
- $\Phi \mid_{p} \mathsf{supp}\,(\alpha) = \varnothing$

Notons que les propriétés d'évolution des attitudes au sujet d'un ensemble croissant (*cf.* section 3.5) peuvent être avantageusement exploitées lors de cette analyse. Par exemple, si Φ est contre une certaine partie P du support, alors la propriété 3.15 affirme que Φ sera contre toute partie de ce support incluant P.

Génération d'arguments avec réponses pertinentes : Comme Φ est pour une partie du support de l'argument qui lui est présenté, la propriété 6.13 lui garantit qu'il peut générer au moins un argument en faveur de ce dernier :

$$\begin{aligned} &\{windows \Rightarrow stable\} \in \Phi \mid_{+} \mathsf{supp}\,(\alpha) \\ &\qquad \Rightarrow \quad \exists R,\ \langle R, windows \Rightarrow stable\rangle \in \mathcal{A}rg_{\Phi}^{\mathsf{déf}(\alpha)} \end{aligned}$$

Construire ces arguments nécessite d'évaluer l'ensemble des réponses de Φ à $\{\neg(windows \Rightarrow stable)\}$ (*cf.* propriété 6.16). On peut préférer ne construire des arguments qu'à partir de réponses pertinentes, auquel cas la propriété 5.14 sur la pertinence sans interdits nous fournit une formule plus pratique que la seule définition d'une réponse :

$$\mathcal{R}p^{A}_{[K,\{\perp\}]} \;=\; \mathsf{min}\left([A,\{\perp\}] \mid_{\not\vdash} K\right)$$

Il suffit d'unifier l'ensemble K aux connaissances de Φ et d'affecter l'ensemble A à $\{\neg(windows \Rightarrow stable)\}$ pour déterminer que $\mathcal{R}p^{A}_{[K,\{\perp\}]}$ est égal à $\big\{\{(linux \vee windows) \Rightarrow stable\}\big\}$. Un premier argument de l'agent Φ défendant α est donc :

$$\langle\{(linux \vee windows) \Rightarrow stable\}, windows \Rightarrow stable\rangle$$

Comme $\mathcal{R}p^{A}_{[K,\{\perp\}]}$ ne contient qu'un seul ensemble, il n'est pas possible de construire d'autres arguments de Φ à partir de réponses pertinentes défendant α.

Génération d'arguments avec réponses non-pertinentes : Si l'on élargit le spectre de recherche aux réponses simples, il est possible de mettre une fois encore à profit les propriétés portant sur l'évolution des attitudes : le lemme 3.13 précise que si un ensemble de formules n'est pas recevable par un agent, alors tout ensemble le contenant ne sera pas non plus recevable par cet agent. Or nous venons de calculer $\mathsf{min}\left([A,\{\perp\}] \mid_{\not\vdash} K\right)$. Par conséquent toute partie de K contenant l'un de ces minimaux est une réponse de $[K,\{\perp\}]$ à A. Ceci nous donne un second argument de Φ défendant α, mais son support n'est donc pas une réponse pertinente :

$$\langle\{linux, (linux \vee windows) \Rightarrow stable\}, windows \Rightarrow stable\rangle$$

Mensonge : L'agent Φ est dorénavant de mauvaise foi. La formule suivante vient d'être ajoutée au sein de son ensemble X d'interdits : $windows \Rightarrow stable$. Donc : $X = \{\bot, windows \Rightarrow stable\}$. Cela ne change en rien son attitude vis-à-vis des parties du support de l'argument de son interlocuteur : il est toujours neutre au sujet de $\{windows\}$, etc. Par ailleurs, la propriété d'inflation des agents (propriété 5.9) assure que les arguments déjà générés par Φ appartiennent toujours à sont ensemble d'arguments. Néanmoins il peut à présent formuler un argument mensonger, car l'interdit ajouté est tel que $X \setminus \overline{K}$ n'est plus vide — ce qui est une condition *sine qua non*, corollaire 5.17 — et fait partie du support de l'argument α :

$$\{\neg(windows \Rightarrow stable)\} \in \mathcal{A}rg_{\Phi}^{\mathsf{att}(\alpha)}$$

Remarque 6.17. On voit là un atout important des X-inférences. Elles permettent la capture d'un phénomène bien singulier autorisant la production par un agent d'arguments soutenant une thèse en complète contradiction avec ses connaissances, et cela tout en maintenant ces dernières — et c'est de toute façon une obligation — consistantes. Il est alors permis à un agent d'être pour — ou contre — une partie de l'argument de son interlocuteur tout en avançant un argument l'attaquant — ou le défendant. Une seule condition donc : posséder un interdit inféré par ses propres connaissances. Si cet interdit n'était pas déjà présent, son ajout ne modifie en rien l'attitude d'un agent vis-à-vis des sous-ensembles de l'argument de son interlocuteur : si ce premier agent pouvait produire un argument venant soutenir celui avancé par le second, il en aura encore la possiblité après ajout de ce type d'interdit, même si cela lui offre dorénavant une alternative, le mensonge. $\diamondsuit$

6.4 Implémentation : Argutia

Nous avons implémenté une partie du processus de génération d'arguments au sein d'un programme Java, nommé Argutia. Il est à même de générer tous les arguments d'un agent, basés sur des réponses pertinentes, que ce soit pour ou contre le support d'un autre argument. Ceci englobe les réponses pertinentes basés exclusivement sur les connaissances de l'agent, sur ses interdits, ou encore les mensonges de cet agent.

6.4.1 Algorithmes

Voici les algorithmes mis en œuvre pour la génération des réponses pertinentes d'un agent $[K, X]$ à un ensemble de formules S :

- Détermination de la recevabilité d'un ensemble de formules E par un agent $[K, X]$:

```
booléen estRecevable (E, [K,X]) {
   pour toute formule f ∈ X \ K̄ {
      si K ∪ E ∪ {¬f} ⊢ ⊥
         retourne faux;
   }
   retourne vrai;
}
```

- Détermination de l'attitude d'un agent $[K, X]$ vis-à-vis d'un ensemble de formules E :

```
attitude CalculAttitude ([K,X],E) {
   si estRecevable (E,[K,X])
      si estRecevable ({¬E},[K,X])
         retourne neutre;
      sinon
         retourne pour;
   sinon
      si estRecevable (E,[K,X])
         retourne contre;
      sinon
         retourne perplexe;
}
```

- Calcul du partitionnement des interdits d'un agent $[K, X]$ en $X \cap \overline{K}$ et $X \setminus \overline{K}$:

```
PartitionnerInterdits ([K,X]) {
   pour tout f ∈ X {
      si K ∪ {¬f} ⊢ ⊥
         ajouter f à X ∩ K̄;
      sinon
```

```
        ajouter f à X \ K̄;
    }
}
```

– Calcul des attitudes de $[K, X]$ vis-à-vis des parties de l'ensemble de formules E (par exemple $\mathcal{P}(E)(neutre)$ contiendra les parties de E pour lesquelles l'agent est neutre) :

```
P(E) PartitionnerParAttitudes ([K,X],E) {
    PartitionnerInterdits ([K,X]);
    pour tout E' ∈ 2^E {
        attitude = CalculAttitude ([K,X \ K̄],E');
        ajout de E' dans P(E)(attitude);
    }
}
```

– Génération de tous les arguments de l'agent $[K, X]$ basés sur des réponses pertinentes attaquant un argument de support S :

```
GénérerArgumentsAttaquant ([K,X],S) {
    P(S) = PartitionnerParAttitudes ([K,X],S);

    // réponses pertinentes sans interdits :
    pour tout E ∈ P(S)(contre ou perplexe) {
        P(K) = PartitionnerParAttitudes ([E,{⊥}],K);
        pour tout R ∈ min(P(K)(contre ou perplexe)) {
            ajouter R à Rp^S_[K,{⊥}];
        }
    }

    // réponses pertinentes sans connaissances :
    pour tout E ∈ P(S)(contre ou perplexe) {
        pour toute formule f ∈ Ū(X \ K̄) {
            P({f}) = PartitionnerParAttitudes ([E,{⊥}],{f});
            ajouter P({f})(contre ou perplexe) à Rp^S_[∅,X];
        }
    }

    // réponses pertinentes connaissances + interdit :
    pour tout E ∈ P(S)(contre ou perplexe) {
```

        pour tout ensemble de formules $K' \in 2\hat{}K$ {
            pour toute formule $x \in X \setminus \overline{K}$ {
                $\mathcal{P}(E)$ = PartitionnerParAttitudes $([K', \{\bot, x\}], E)$;
                si $\mathcal{P}(E)(\textit{contre ou perplexe})$ est non vide
                    ajouter $K' \cup \{\neg x\}$ à $\mathcal{R}p^{S}_{[K,X]}$;
            }
        }

    // mensonges :
    pour tout $E \in \mathcal{P}(S)(\textit{contre ou perplexe})$ {
        pour tout ensemble de formules $K' \in 2\hat{}K$ {
            pour toute formule $x \in X \cap \overline{K}$ {
                $\mathcal{P}(E)$ = PartitionnerParAttitudes $([K', \{\bot, x\}], E)$;
                si $\mathcal{P}(E)(\textit{contre ou perplexe})$ est non vide
                    ajouter $K' \cup \{\neg x\}$ à $\mathcal{R}m^{\{\neg S\}}_{[K,X]}$;
            }
        }
    }

6.4.2 Complexité

Afin d'illustrer la complexité des algorithmes mis en jeu dans Argutia, prenons l'exemple du calcul des réponses d'un agent $[K, X]$ à un argument α :

1. pour tout $E \in [K, X] \,|_{\not\vdash}\, \mathsf{supp}\,(\alpha)$	$2\hat{}|\mathsf{supp}\,(\alpha)| \times |X| \times 2\hat{}n$
2. pour tout $K' \subseteq K$	$2\hat{}|K|$
3. pour tout $x \in X \setminus \overline{K'}$	$|X \setminus \overline{K'}|$
4. si $K' \cup \{\neg x\} \cup E \vdash \bot$	$2\hat{}n'$
5. ajouter $K' \cup \{\neg x\}$ à $\mathcal{R}^{\mathsf{supp}(\alpha)}_{[K,X]}$	

À droite de l'algorithme figurent les complexités associées :
- la ligne 4 s'effectue en $\mathcal{O}(2^{n'})$, où n' est le nombre de variables propositionnelles de $K' \cup \{\neg x\} \cup E$;
- la troisième ligne boucle autant de fois qu'il y a de formules dans $X \setminus K'$;
- la ligne 2 calcule tous les sous-ensembles de K et est donc en $\mathcal{O}(2^{|K|})$;
- enfin, la première ligne s'effectue en $\mathcal{O}(2^{|\mathsf{supp}(\alpha)|} \times |X| \times 2^n)$, où n est le nombre de variables propositionnelles de $K \cup X \cup E$, $E \subseteq \mathsf{supp}\,(\alpha)$.

La ligne 1 indique que nous appartenons au moins à la classe $\sum_2^P$, deuxième niveau dans la hiérarchie polynômiale, comme tout autre formalisme non-monotone.

6.4.3 Présentation

La figure 6.1 montre Argutia une fois chargé avec l'exemple 3.9. Le logiciel est alors positionné sur un onglet intitulé « Connaissances ». On y voit plusieurs zones de texte symbolisant divers ensembles de formules : « K_A » et « X_A » représentent les connaissances et interdits d'un agent « A » ; « support de l'argument » représente l'ensemble de formules support de l'argument présenté à l'agent ; enfin la zone « X_B » ne sert pas encore et n'est là que parce qu'à terme nous éspérons pouvoir simuler les divers échanges argumentatifs possibles entre deux agents. À noter que la contradiction est sytématiquement ajoutée à l'ensemble des interdits de l'agent.

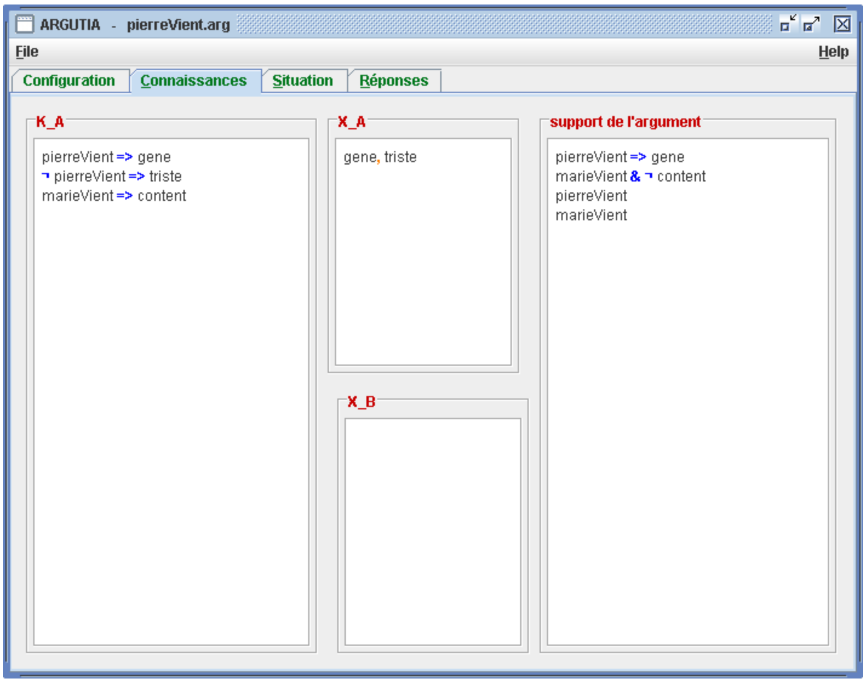


FIG. 6.1 – Onglet « Connaissances » d'Argutia, chargé avec l'exemple 3.9

L'onglet « Situation » (figure 6.2) permet de lancer le calcul et de visionner les attitudes de l'agent à propos des sous-ensembles du support avancé. Y figure le partitionnement des interdits X en $X \setminus \overline{K}$ et $X \cap \overline{K}$, puis la détermination des formules du support à propos desquelles l'agent est pour, neutre, contre et perplexe, et enfin le partitionnement de l'ensemble des parties du support en fonction des attitudes de l'agent à leur propos (ces partitions sont notées « Pour* », « Neutre* », « Contre* » et « Perplexe* »).

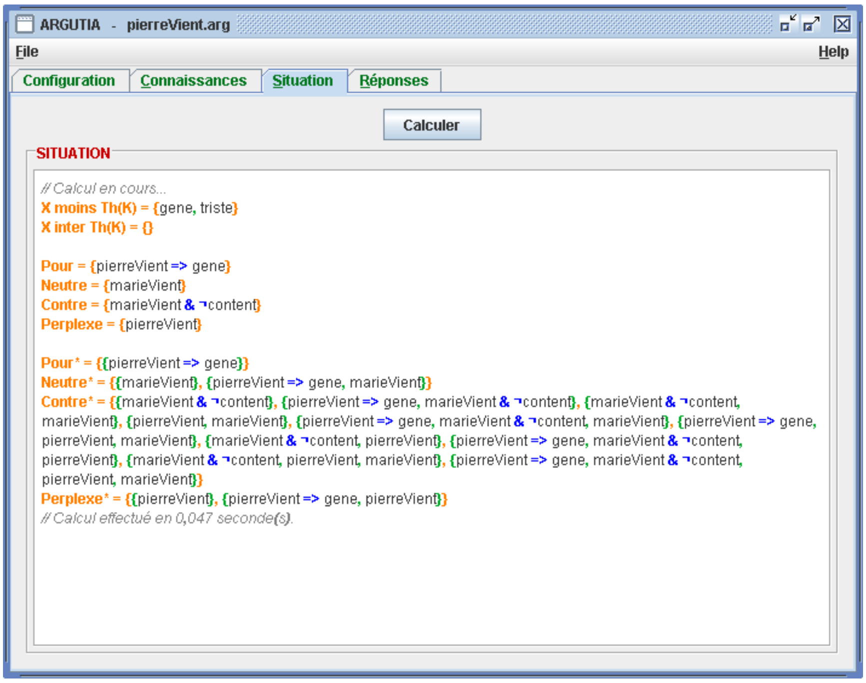


FIG. 6.2 – Onglet « Situation » d'Argutia, chargé avec l'exemple 3.9

Une fois ces ensembles calculés, le dernier onglet intitulé « Réponses » (figure 6.3) peut être activé afin de lancer le processus de génération d'arguments. Seuls sont générés les arguments de l'agent dont le support est une réponse pertinente, qu'ils soient ou non en faveur du support de l'argument présenté à l'onglet « Connaissances ». Dans l'ordre sont énumérés :

- les arguments attaquant le support présenté et basés
  - uniquement sur les connaissances de l'agent,
  - sur un mensonge,
  - sur un interdit de l'agent,
  - ou encore utilisant à la fois les connaissances et les interdits ;
- puis viennent les arguments défendant le support présenté, et basés là aussi
  - uniquement sur les connaissances de l'agent,
  - sur un mensonge,
  - sur un interdit de l'agent,
  - ou enfin utilisant à la fois les connaissances et les interdits.

Comme l'agent est perplexe au sujet de la formule « PierreVient », Argutia propose à la fois un argument pour et contre cette formule :

- $\langle\{\neg pierreVient \Rightarrow triste, \neg triste\}, pierreVient\rangle$
- $\langle\{pierreVient \Rightarrow gene, \neg gene\}, \neg(pierreVient)\rangle$

L'agent est pour la formule « PierreVient $\Rightarrow$ gene », en effet :

- $\langle\{pierreVient \Rightarrow gene\}, pierreVient \Rightarrow gene\rangle$

Enfin, si l'agent est contre la formule « marieVient $\wedge\neg$ content », c'est qu'il possède le contre-argument suivant :

- $\langle\{marieVient \Rightarrow content\}, \neg(marieVient \wedge \neg content)\rangle$

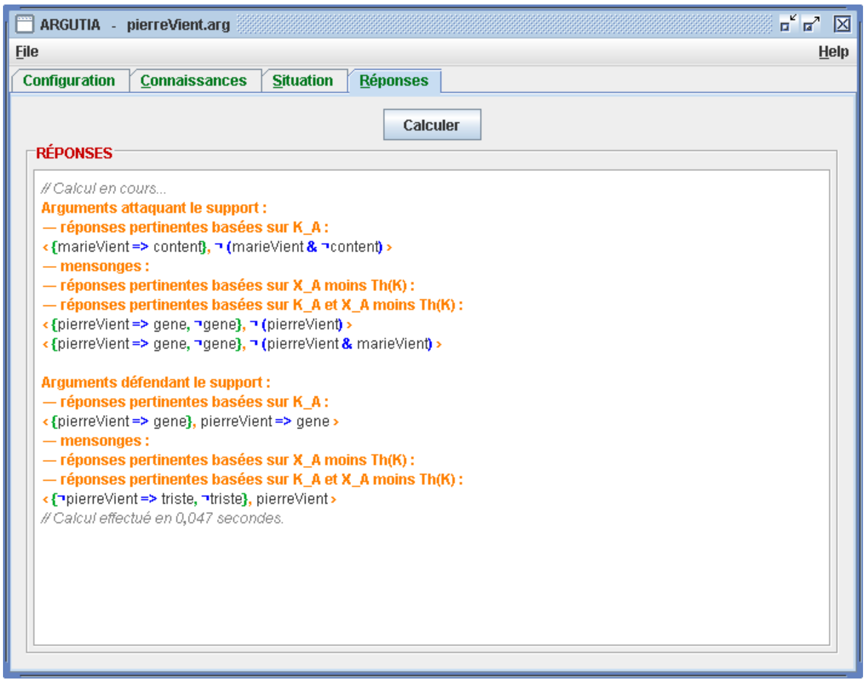


FIG. 6.3 – Onglet « Réponses » d'Argutia, chargé avec l'exemple 3.9

Chapitre 7

Application : contre-arguments conservatifs maximaux

Afin de réunir arguments et contre-arguments pour ou contre une thèse initiale, Besnard et Hunter construisent dans [BH01] des arbres argumentatifs au sein desquels s'imbriquent ces arguments, simulant de fait un débat. Le regroupement dans une structure argumentative de tous les arbres dont la racine statue sur un fait particulier permet alors de mesurer le crédit que l'on peut accorder à cette thèse (voir section 1.6.1 sur les catégoriseurs et accumulateurs).

Ces arbres ont ceci de particulier qu'ils ne contiennent que des *contre-arguments conservatifs maximaux* (Ccm), arguments retenus par Besnard et Hunter pour leur pertinence. Nous allons montrer comment générer de tels arguments.

7.1 Rappels

Définition 7.1 (Argument minimal, [BH01]). *Soit Δ un ensemble fini et éventuellement inconsistant de formules représentant les connaissances et croyances du système. Un* argument minimal *est un couple $\langle S, c\rangle$ tel que :*

1. $S \nvdash \bot$
2. *S est un sous-ensemble minimal de Δ tel que $S \vdash c$.*

Définition 7.2 (Contre-argument, [BH01])**.** *Un* contre-argument *pour un argument minimal α est un argument minimal $\langle B, \neg(a_1 \wedge \cdots \wedge a_n)\rangle$ où $\{a_1, \ldots, a_n\} \subseteq \mathsf{supp}(\alpha)$.*

Définition 7.3 (Conservativité, [BH01])**.** *Un argument α est* plus conservatif *qu'un argument β si et seulement si le support de α est inclus dans le support de β et la conclusion de β infère celle de α.*

Définition 7.4 (CCM, [BH01])**.** *Un argument β est un* contre-argument conservatif maximal *pour l'argument α si et seulement si β est un contre-argument pour α tel qu'aucun autre contre-argument pour α est strictement plus conservatif que β.*

Propriété 7.5 ([BH01])**.** *Si $\langle B, \neg(a_1 \wedge \cdots \wedge a_n)\rangle$ est un* contre-argument conservatif maximal *pour l'argument minimal α, alors : $\mathsf{supp}(\alpha) = \{a_1, \ldots, a_n\}$.*

7.2 Génération de contre-arguments conservatifs maximaux

Les arguments de Besnard et Hunter satisfont une contrainte de minimalité du support (définition 7.1). Dans ce cadre, la notion de contrariété entre arguments (définition 6.3) ne peut plus survenir et par conséquent toute relation d'attaque entre arguments peut se ramener à celle de réfutation.

Pour retrouver le concept d'agent, il faut imaginer que les connaissances contenues dans l'ensemble Δ appartenaient avant fusion à deux agents. Comme la notion d'interdits n'existe pas dans ce système argumentatif, nos agents ont leur ensemble d'interdits réduit à $\{\bot\}$.

Nous cherchons à exprimer la notion de contre-argument conservatif maximal au sein de notre système. Elle n'est donc plus relative à un ensemble Δ mais fonction des connaissances de l'agent.

Définition 7.6 (CCM d'un agent)**.** *Un argument α est un* CCM *de l'agent $[K, \{\bot\}]$ pour l'argument β si et seulement si α est un* CCM *pour β et α appartient aux arguments de l'agent $[K, \{\bot\}]$.*

Propriété 7.7. *Un argument α est un* CCM *de l'agent $[K, \{\bot\}]$ pour l'argument β si et seulement si :*

1. $\mathsf{supp}(\alpha) \in \mathcal{R}p^{\mathsf{supp}(\beta)}_{[K,\{\bot\}]}$
2. $\mathsf{concl}(\alpha) = \neg\mathsf{supp}(\beta)$

Démonstration.

(⇒) L'argument α est un Ccm pour l'argument β, donc, par la propriété 7.5, nous obtenons : $\mathsf{concl}\,(\alpha) = \neg\mathsf{supp}\,(\beta)$. Par ailleurs, α est un argument de l'agent $[K, \{\bot\}]$, par conséquent son support appartient aux réponses de $[K, \{\bot\}]$ à $\neg\mathsf{concl}\,(\alpha)$. Comme α est un Ccm, alors : $\nexists E \subset \mathsf{supp}\,(\alpha)\,,\ E \vdash \mathsf{concl}\,(\alpha)$. Donc si l'on applique la définition d'une réponse pertinente (définition 5.10), on obtient que : $\mathsf{supp}\,(\alpha) \in \mathcal{R}p^{\neg\mathsf{concl}(\alpha)}_{[K,\{\bot\}]}$.

(⇐) Par hypothèse, $\mathsf{supp}\,(\alpha)$ appartient aux réponses pertinentes de l'agent $[K, \{\bot\}]$ au support de β. Il en résulte : $\nexists E \subset \mathsf{supp}\,(\alpha)\,,\ E \vdash \mathsf{concl}\,(\alpha)$. Ainsi, et d'après la définition 7.4, l'argument $\langle \mathsf{supp}\,(\alpha)\,, \neg\mathsf{supp}\,(\beta)\rangle$ est un Ccm de l'argument β. Comme la conclusion de α est la négation du support de β, α est un Ccm de β.

□

Corollaire 7.8 (Génération de Ccm)**.** *Un argument α est un* Ccm *de l'agent $[K, \{\bot\}]$ pour l'argument β si et seulement si :*

1. $\mathsf{supp}\,(\alpha) \in \mathsf{min}\left([\mathsf{supp}\,(\beta)\,, \{\bot\}] \,|_{\nvdash}\, K\right)$
2. $\mathsf{concl}\,(\alpha) = \neg\mathsf{supp}\,(\beta)$

Démonstration. Application directe de la propriété 7.7 puis de celle intitulée pertinence sans interdits (propriété 5.14). □

Exemple 7.9. Reprenons l'arbre argumentatif présenté dans l'exemple 1.28. Soit $\Delta = \{a \Leftrightarrow \neg d, b, b \Rightarrow a, c \wedge \neg b, \neg c, d, \neg d\}$ l'ensemble de connaissances commun. À partir de cet ensemble il est possible de construire l'arbre suivant, en faveur de la formule a :

$$\begin{array}{ccccc}
 & & \langle\{b \Rightarrow a, b\}, a\rangle & & \\
 & \nearrow & & \nwarrow & \\
\langle\{c \wedge \neg b\}, \diamond\rangle & & & & \langle\{d, a \Leftrightarrow \neg d\}, \diamond\rangle \\
\uparrow & & & & \uparrow \\
\langle\{\neg c\}, \diamond\rangle & & & & \langle\{\neg d\}, \diamond\rangle
\end{array}$$

Pour résoudre le calcul des contre-arguments conservatifs maximaux pour la racine de l'arbre, à savoir l'argument $\langle\{b \Rightarrow a, b\}, a\rangle$, nous considérons les deux agents suivants :

- $\Phi = [\{a \Leftrightarrow \neg d, b, b \Rightarrow a, \neg c, \neg d\}, \{\bot\}]$
- $\Psi = [\{a \Leftrightarrow \neg d, b \Rightarrow a, c \wedge \neg b, d\}, \{\bot\}]$

L'agent Φ a proposé l'argument $\langle\{b \Rightarrow a, b\}, a\rangle$ et l'agent Ψ doit donc générer des Ccm pour cet argument. Le corollaire 7.8 sur la génération de Ccm permet de déduire que les Ccm recherchés sont tels que leur support est à prendre dans : $\mathsf{min}\left([\{b \Rightarrow a, b\}, \{\bot\}] \mid_{\not\vdash} K_\Psi\right)$, où $K_\Psi = \{a \Leftrightarrow \neg d, b \Rightarrow a, c \wedge \neg b, d\}$.

$$\mathsf{min}\left([\{b \Rightarrow a, b\}, \{\bot\}] \mid_{\not\vdash} K_\Psi\right) = \big\{\{a \Leftrightarrow \neg d, d\}, \{c \wedge \neg b\}\big\}$$

Par conséquent, les Ccm de l'agent Ψ pour l'argument $\langle\{b \Rightarrow a, b\}, a\rangle$ sont les deux arguments suivants :
- $\langle\{a \Leftrightarrow \neg d, d\}, \neg\{b \Rightarrow a, b\}\rangle$
- $\langle\{c \wedge \neg b\}, \neg\{b \Rightarrow a, b\}\rangle$

Nous obtenons ainsi les deux nœuds fils de l'arbre argumentatif. Les nœuds suivants sont générés de manière similaire. $\diamondsuit$

Synthèse et perspectives

À partir de la définition et des propriétés de la X-inférence, un nouveau cadre argumentatif a été proposé :

- Les X-logiques n'avaient pas encore été utilisées au sein d'un système argumentatif. La notion de *recevabilité* caractéristique des X-logiques nous a permis en premier lieu d'élaborer le concept d'*attitudes*, pierre angulaire de notre approche. Ce concept repose sur la paraconsistance des X-inférences dans le cas général. Il est à ce propos remarquable de constater que le plan de l'octaèdre qui définit les attitudes est en correspondance directe avec le bi-treillis de la logique $\mathcal{FOUR}$ de Belnap.

- Les attitudes sont exploitées par des *agents* qui pour le coup sont vus comme l'association d'une base de connaissances et d'un ensemble de formules interdites en relation avec l'ensemble X de la X-inférence.

- Les *opérateurs de confrontation* permettent alors le partitionnement d'un ensemble de formules en fonction des attitudes d'un agent vis-à-vis de cet ensemble. Un argument étant vu comme l'association d'un support de formules et d'une conclusion, ces opérateurs permettent l'analyse d'un argument présenté à un agent. Ainsi les attitudes, au travers des opérateurs de confrontation, permettent d'atteindre une granularité plus fine dans l'étude des relations de conflit entre arguments : un agent est en mesure de se prononcer à propos des formules-même du support d'un argument.

– Nous nous sommes ensuite intéressés au processus de génération d'un argument à partir d'un ensemble de formules et des attitudes d'un agent face à cet ensemble. Notre approche lie agent et arguments via la notion de *réponses*, désignant les ensembles de formules qui « motivent » les attitudes de l'agent. Trois niveaux de réponses sont distingués : un niveau de base correspondant à la définition générale, un niveau plus fin associé aux réponses *pertinentes* et un niveau encore plus fin associé aux *mensonges*.

– La redéfinition d'un argument à partir de la notion de réponse à montré qu'à chaque ensemble produit par les opérateurs de confrontation correspond au moins un argument potentiel. Cette approche est contextuelle : elle rend tout argument potentiel dépendant du contexte définit par l'argument précédent.

– Nous avons finalement présenté une application de notre système argumentatif en s'attachant à générer tous les *contre-arguments conservatifs maximaux* des arbres de Besnard et Hunter, arguments retenus par ces derniers pour leur pertinence.

Bien entendu, de nombreuses questions restent en suspens. En premier lieu, la question du rapprochement entre les attitudes d'un agent et la logique multi-valuée de Belnap doit être étudiée. Une des difficultés provient du fait qu'il n'est pas possible de reconstruire directement la fonction de valuation de la logique $\mathcal{FOUR}$, et par conséquent de combiner nos arguments à l'aide des connecteurs de cette logique. La raison est que l'attitude d'un agent au sujet de l'union de deux ensembles n'est pas systématiquement déterminée de manière univoque par l'attitude de cet agent au sujet de chacun des ensembles.

D'autre part les attitudes d'un agent concernent les parties du support d'un argument. On peut imaginer élaborer des stratégies visant à déterminer une attitude de l'agent par rapport à l'argument lui-même, aboutissant à une notion d'acceptabilité individuelle telle qu'introduite notamment par Elang-Gøransson, Fox ou Hunter. Par exemple si l'agent est contre une partie d'un argument, les propriétés d'évolution des attitudes prédisent qu'il sera contre le support entier de cet argument. Mais si l'agent se trouve être pour une autre partie de ce support, il peut ne pas être souhaitable de maintenir l'agent contre l'argument.

Nous n'avons pas abordé la question des classes d'acceptabilité conjointes telles que définies principalement par Dung. Soulignons toutefois que notre propos a trait à la génération dynamique d'arguments dans un cadre dialogique et ne cherche donc pas à représenter la discussion elle-même, étape nécessaire à la définition de telles classes d'acceptabilité.

D'autres types de réponses peuvent être envisagés. Nous pensons entre autre à des réponses « nouvelles » qui garantiraient l'apport de formules non encore utilisées au sein de la discussion, ainsi qu'à des réponses « détaillées », sous-ensemble des réponses pertinentes maximisant la quantité d'information.

Enfin, lors de la caractérisation des réponses nous recourons à des agents « virtuels » créés pour le propos. On cherche à déterminer l'attitude de ces agents au sujet des connaissances ou de certains interdits de l'agent de départ. En quelque sorte l'agent argumente avec lui-même. Cette idée laisse envisager la possibilité d'une étude formelle de l'introspection au sein de notre cadre argumentatif.

Bibliographie

[AC98] Leila Amgoud and Claudette Cayrol. On the acceptability of arguments in preference-based argumentation. In *UAI*, pages 1–7, 1998. 23, 28, 29, 45, 83, 85

[AD85a] Stephanie E. August and Michael G. Dyer. Analogy recognition and comprehension in editorials. In *7th Annual Conference of the Cognitive Science Society*, pages 228–235, 1985. 12

[AD85b] Stephanie E. August and Michael G. Dyer. Understanding analogies in editorials. In *IJCAI : 9th International Joint Conference on Artificial Intelligence*, pages 845–847, 1985. 12

[AR05] G. Aubry and V. Risch. Toward a logical tool for generating new arguments in an argumentation based framework. In *Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2005)*, Hong Kong, China, 14–16 november 2005. IEEE Computer Society. ISBN : 0–7695-2488–5. 13, 39

[AR06] G. Aubry and V. Risch. Génération d'arguments au sein d'un système argumentatif. In *15e Congrès Reconnaissance des Formes et Intelligence Artificielle*, Tours, France, janvier 2006. à paraître. 13, 39, 80

[Aub02] G. Aubry. Sur la génération d'arguments à l'aide des x-logiques. In *Journées des Doctorants du LSIS (JDL6)*, pages 13–21, Marseille, France, décembre 2002. 13, 39

[BDP93] Salem Benferhat, Didier Dubois, and Henri Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In David Heckerman and E. H. Mamdani, editors, *UAI '93 : Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence, July 9-11, 1993, The Catholic University of America, Providence, Washington, DC, USA*, pages 411–419. Morgan Kaufmann, 1993. 29

[Bel77] Jr. Belnap, N. D. A useful four-valued logic. In Michael Dunn and G. Epstein, editors, *Modern Uses of Multiple-Valued Logic*, pages 8–37. Reidel, Dordrecht, 1977. 51

[BH01] Philippe Besnard and Anthony Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128(1-2) :203–235, 2001. 14, 23, 28, 35, 36, 37, 38, 45, 83, 99, 100

[Bob80] D. G. Bobrow. Special issue on non-monotonic logic. *Artificial Intelligence*, 13, 1980. 11, 39

[Boc03] Alexander Bochman. Brave nonmonotonic inference and its kinds. *Annals of Mathematics and Artificial Intelligence*, 39(1–2) :101–121, 2003. 42

[Bre94] G. Brewka. A reconstruction of rescher's theory of formal disputation based on default logic. In *Proc. of the 11th ECAI*, pages 366–370, Amsterdam, The Netherlands, 1994. 28

[Che96] Carlos I. Chesñevar. El Problema de la Inferencia en Sistemas Argumentativos : Alternativas para su Solución (MSc Thesis). Master's thesis, Departamento de Cs. de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, December 1996. 34

[CLS94] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the complexity of non-monotonic entailment in syntax-based approaches. In Marco Schaerf, editor, *Proceedings ECAI'94 Workshop on Algorithms, Complexity and Commonsense Reasoning*, Amsterdam, 1994. 28

[CLS05] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Gradual valuation for bipolar argumentation frameworks. In Lluis Godo, editor, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 8th European Conference, ECSQARU 2005, Barcelona, Spain*, volume 3571 of *Lecture Notes in Computer Science*, pages 366–377. Springer, 2005. 34

[CML00] Carlos Iván Chesñevar, Ana Gabriela Maguitman, and Ronald Prescott Loui. Logical models of argument. *ACM Computing Surveys*, 32(4) :337–383, 2000. 12, 13, 27, 28

[DFK96] Subrata Kumar Das, John Fox, and Paul Krause. A unified framework for hypothetical and practical reasoning (1) : Theoretical foundations. In *FAPR '96 : Proceedings of the International Conference on Formal and Applied Practical Reasoning*, pages 58–72, London, UK, 1996. Springer-Verlag. 12

[Dun93a] Phan Minh Dung. An argumentation semantics for logic programming with explicit negation. In *ICLP*, pages 616–630, 1993. 28

[Dun93b] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning and logic programming. In *IJCAI : 13th International Joint Conference on Artificial Intelligence*, pages 852–859, 1993. 28, 31

[Dun95] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2) :321–358, 1995. 21, 28, 31, 45, 85

[EGH95] Morten Elvang-Gøransson and Anthony Hunter. Argumentative logics : Reasoning with classically inconsistent information. *Data Knowl. Eng.*, 16(2) :125–145, 1995. 30

[EGKF93] Morten Elvang-Gøransson, Paul Krause, and John Fox. Dialectic reasoning with inconsistent information. In David Heckerman and E. H. Mamdani, editors, *UAI '93 : Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence, The Catholic University of America, Providence, Washington, DC, USA*, pages 114–121. Morgan Kaufmann, July 9–11 1993. 23, 29, 30, 83

[FAM96] George Ferguson, James F. Allen, and Brad Miller. Trains-95 : Towards a mixed-initiative planning assistant. In *AIPS : 3rd International Conference on AI Planning Systems*, pages 70–77, 1996. 12

[Fit91] Melvin Fitting. Bilattices and the semantics of logic programming. *J. Log. Program.*, 11(1&2) :91–116, 1991. 51

[FS98] Lionel Forget and Pierre Siegel. A proof procedure for circumscription. In *Proceedings of the KR'98 Workshop on Computa-*

tional Aspects of Nonmonotonic Reasoning, pages 19–28, 1998. 44

[Gar97] Alejandro J. García. La programación en lógica rebatible, su definición teórica y computacional. Master's thesis, Universidad Nacional del Sur, Bahía Blanca, Argentina, 1997. 34

[GGTS00] A. Garcia, D. Gollapally, P. Tarau, and G. Simari. Deliberative stock market agents using jinni and defeasible logic programming. In *ECAI Workshop on Engineering Societies in the Agents' World*, Berlin, Germany, august 2000. Springer Verlag. 12

[Gor95] Thomas F. Gordon. *Pleadings Game : An Artificial Intelligence Model of Procedural Justice*. Kluwer Academic Publishers, Norwell, MA, USA, 1995. 12

[Gro91] Benjamin N. Grosof. Generalizing prioritization. In *KR*, pages 289–300, 1991. 28

[GS03] Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming : An argumentative approach. *CoRR (The Computing Research Repository)*, cs.AI/0302029, 2003. 28

[KP89] Kurt Konolige and Martha E. Pollack. Ascribing plans to agents. In *IJCAI : 11th International Joint Conference on Artificial Intelligence*, pages 924–930, Detroit, USA, 1989. 12

[KP93] Kurt Konolige and Martha E. Pollack. A representationalist theory of intention. In *IJCAI : International Joint Conference on Artificial Intelligence*, pages 390–395, 1993. 12

[LNOM93] Ronald P. Loui, Jeff Norman, Jon Olson, and Andrew Merrill. A design for reasoning with policies, precedents, and rationales. In *ICAIL '93 : Proceedings of the 4th international conference on Artificial intelligence and law*, pages 202–211, New York, NY, USA, 1993. ACM Press. 12

[Lou87] R. P. Loui. Defeat among arguments : a system of defeasible inference. *Computational Intelligence*, 3 :100–106, 1987. 20, 26

[LS89] Fangzhen Lin and Yoav Shoham. Argument systems : A uniform basis for nonmonotonic reasoning. In Morgan Kaufmann, editor, *KR'89*, pages 245–255, 1989. 20, 21, 23, 24, 45

[MH87] J. McCarthy and P. J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. *Readings in nonmonotonic reasoning*, pages 26–45, 1987. 11

[Pol70] John L. Pollock. The structure of epistemic justification. *American Philosophical Quarterly*, monograph series 4 :62–78, 1970. 28

[Pol87] John L. Pollock. Defeasible reasoning. *Cognitive Science*, 11(4) :481–518, 1987. 20, 22

[Pol92] John L. Pollock. How to reason defeasibly. *Artificial Intelligence*, 57(1) :1–42, 1992. 32

[Poo88] David Poole. A logical framework for default reasoning. *Artificial Intelligence*, 36(1) :27–47, 1988. 26

[Pra93] Henry Prakken. An argumentation framework in default logic. *Ann. Math. Artif. Intell.*, 9(1-2) :93–132, 1993. 12

[Pra97] Henry Prakken. Logical tools for modelling legal argument : A study of defeasible reasoning in law. *Actes de 4th ModelAge Workshop on formal Models of Agents*, pages 201–214, 1997. 12, 26

[PS96] H. Prakken and G. Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artificial Intelligence and Law*, 4 :331–368, 1996. 12, 20, 22, 26

[PS97] Henry Prakken and Giovanni Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7(1), 1997. 22, 31

[PSJ98] Simon Parsons, Carles Sierra, and Nick Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3) :261–292, 1998. 12

[PV02] H. Prakken and G. Vreeswijk. Logical systems for defeasible argumentation. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 4, pages 219–318. Kluwer Academic, Dordrecht, 2nd edition, 2002. 28, 31

[Rei78] R. Reiter. On closed-word data bases. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 55–76. Plenum Press, New York, 1978. 46, 50

[Rei80] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1-2) :81–132, 1980. 20

[Res77] N. Rescher. *Dialectics, A Controversy-Oriented Approach to the Theory of Knowledge*. State University of New York Press, Albany., 1977. 18, 19, 45

[Sad00] S. Sadok. Une caractérisation des X-logiques en termes de tableaux analytiques. *Mémoire de DEA*, juin 2000. 41

[Sar94] G. Sartor. A formal model of legal argumentation. *Ratio Juris 7*, pages 212–226, 1994. 12

[SCG94] Guillermo R. Simari, Carlos I. Chesñevar, and Alejandro J. García. The role of dialectics in defeasible argumentation. In *Anales de la XIV Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación*. Universidad de Concepción, Concepción (Chile), nov 1994. 33

[SF96] Pierre Siegel and Lionel Forget. A representation theorem for preferential logics. In *KR'96, Proceedings of the fifth International Conference on the Principles of Knowledge Representation and Reasoning, KR'96*, pages 453–460. Morgan Kaufmann, 1996. 13, 39, 40, 42, 43

[SL92] Guillermo R. Simari and Ronald P. Loui. A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence*, 53(2-3) :125–157, 1992. 20, 23, 25, 26, 32, 45, 83

[Suc93] Marek A. Suchenek. First-order syntactic characterization of minimal entailment, domain minimal entailment, and herbrand entailment. *Journal of Automated Reasoning*, 10(2) :237–263, 1993. 44

[Syc89] Katia P. Sycara. Argumentation : Planning other agents' plans. In *IJCAI : 11th International Joint Conference on Artificial Intelligence*, pages 517–523, Detroit, USA, 1989. 12

[Syc90] Katia Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28(3) :203–242, May 1990. 12

[Vre91] Gerard Vreeswijk. The feasibility of defeat in defeasible reasoning. In *KR*, pages 526–534, 1991. 20, 21

[Vre92] G. Vreeswijk. Reasoning with defeasible arguments : Examples and applications. In D. Pearce and G. Wagner, editors, *Logics in AI : Proc. of the European Workshop JELIA'92*, pages 189–211. Springer, Berlin, Heidelberg, 1992. 25, 32, 45

Un cadre logique pour la génération d'arguments

Résumé. L'idée de l'argumentation est de rechercher dans une base de connaissances, pour chaque proposition dont on souhaite évaluer la validité, les raisons qui étayent cette proposition et celles qui l'infirment. Un argument est alors entendu comme une paire comprenant une proposition et les raisons qui la justifient. Notre propos est d'offrir des outils formels pour la génération automatique d'arguments par deux agents en situation de dialogue. Ces outils reposent sur les X-logiques, formalisme non-monotone proposé en 1996 par Siegel et Forget et déterminant un cadre fondateur autour de la notion de preuve pour le raisonnement non-monotone. En particulier l'ensemble X servant à paramétrer la relation d'inférence confère une souplesse inégalée à la gestion dynamique des arguments.

Après un tour d'horizon des travaux passés en matière de représentations logiques pour l'argumentation, nous introduisons les X-logiques, à partir desquelles est composée la notion d'attitude d'un agent par rapport à une formule. Nous définissons ensuite des opérateurs de confrontation qui permettent d'associer des ensembles de formules aux attitudes d'un agent. Le concept de réponse d'un agent à un ensemble de formules est alors élaboré en tant que motivation de l'attitude de cet agent vis-à-vis de l'ensemble en question. Plusieurs formes de réponses sont distinguées parmi lesquelles les notions de réponse pertinente ou encore de mensonge. Une réponse représente les raisons qui justifient la conclusion d'un argument : c'est à partir du calcul de ces réponses que nous exhibons une procédure de génération automatique d'arguments. Enfin nous montrons que notre cadre argumentatif permet de générer les contre-arguments conservatifs maximaux de Besnard et Hunter (2001), arguments retenus pour leur pertinence.

Mots-clefs. Représentation des connaissances, systèmes argumentatifs, arguments, raisonnement, croyances, non-monotonie, X-logiques.

A Logical Framework for the Generation of Arguments

Abstract. The purpose of Argumentation is, in a knowledge base, for every proposition which validity has to be tested, to look for the reasons that support this proposition, as well as for the reasons that invalidate it. To this extent, a proposition is understood as a pair composed of both a proposition and the reason by which this proposition is meant to follow. Our purpose is to construct formal tools that allow the automatic generation of arguments by two agents having to face their respective knowledge. These tools rely on X-logics, a nonmonotonic formalism proposed by Siegel and Forget in 1996 as a general framework for achieving proofs in nonmonotonic reasoning. We show that the set X used to define the parameter of the inference relation gives the dynamic treatment of arguments an incomparable flexibility.

After a survey of existing works related to logics and argumentation, X-logics are introduced, from which is composed the notion of attitude of an agent regarding a formula. Confrontation operators that map sets of formulas with the attitudes of the agent are then proposed. Further, a notion of answer that motivates the attitudes of the agent regarding a set of formulas is defined. Different forms of answers are distinguished, among which the notions of relevant answer, and lie. An answer representing the reasons that justify the conclusion of an argument, the computation of answers provides us with a method of automatic generation of arguments. Finally, we show that our argumentative framework can handle the generation of maximal conservative undercuts of Besnard and Hunter (2001), a kind of arguments especially interesting regarding relevance.

Keywords. Knowledge representation, argumentation systems, arguments, reasoning, beliefs, nonmonotonicity, X-logics.